

VYSOKÁ ŠKOLA BÁŇSKÁ – TECHNICKÁ UNIVERZITA OSTRAVA
EKONOMICKÁ FAKULTA

KATEDRA APLIKOVANÉ INFORMATIKY

Databázová aplikace pro maloobchodní jednotku
The Database Application for the Retail Unit

Student: Kateřina Martynková
Vedoucí bakalářské práce: Ing. Vítězslav Novák, Ph.D.

Ostrava 2014

VŠB - Technická univerzita Ostrava
Ekonomická fakulta
Katedra aplikované informatiky

Zadání bakalářské práce

Student:

Kateřina Martynková

Studijní program:

B6209 Systémové inženýrství a informatika

Studijní obor:

6209R001 Aplikovaná informatika

Téma:

Databázová aplikace pro maloobchodní jednotku
The Database Application for the Retail Unit

Zásady pro vypracování:

1. Úvod
2. Přehled teoretických pojmů
3. Náhled současné situace v podniku
4. Návrh databázové aplikace a implementace
5. Závěr

Seznam použité literatury

Seznam zkratk

Prohlášení o využití výsledků bakalářské práce

Seznam příloh

Přílohy

Seznam doporučené odborné literatury:

CONOLLY, T., R. HOLOWCZAK a C. BEGG. *Mistrovství - databáze: profesionální průvodce tvorbou efektivních databází*. Přeložil Vilém GUTFREUND. Brno: Computer Press, 2009.

ISBN 978-80-251-2328-7.

SHEPHERD, Richard. *Access VBA : výukový průvodce*. Přeložil Jakub MUŽÍK. Brno: Computer Press, 2012. ISBN 978-80-251-3686-7.

VYBÍHAL, V., J. PŘIB a T. BERNREITER. *Mzdové účetnictví 2013: praktický průvodce*. Praha: Grada, 2013. ISBN 978-80-247-4627-2.

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **Ing. Vítězslav Novák, Ph.D.**

Datum zadání: 22.11.2013

Datum odevzdání: 09.05.2014

Ing. Petr Rozehnal, Ph.D.
vedoucí katedry

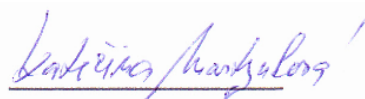


prof. Dr. Ing. Dana Dluhošová
děkanka fakulty

Místopřísežné prohlášení o samostatném vypracování bakalářské práce

„Prohlašuji, že jsem celou práci, včetně všech příloh, vypracovala samostatně“.

Zároveň bych na tomto místě chtěla poděkovat svému vedoucímu práce Ing. Vítězslavu Novákovi, Ph.D. za vedení bakalářské práce, připomínky a rady. Dále také celé své rodině za možnost studovat a příteli za velkou podporu.



Kateřina Martynková

Datum odevzdání bakalářské práce: 9. května 2014

Obsah

1. Úvod.....	5
2. Přehled teoretických pojmů.....	7
2.1. Návrh databáze.....	7
2.1.1. Databáze.....	7
2.1.2. Systém řízení báze dat	8
2.1.3. Třívrstvá architektura databázových systémů.....	9
2.1.4. Relační databáze	11
2.1.5. Komponenty relační databáze.....	11
2.1.6. Normalizace	16
2.1.7. Bezpečnost	17
2.2. Microsoft Access.....	18
2.2.1. Tabulka	18
2.2.2. Datové typy a vlastnosti.....	18
2.2.3. Formuláře.....	20
2.2.4. Sestavy	20
2.2.5. Výběr dat.....	21
2.2.6. Visual Basic for Application (VBA)	23
2.3. Mzdové účetnictví	25
2.3.1. Hrubá a čistá mzda.....	25
2.3.2. Zdravotní pojištění.....	26
2.3.3. Pojistné na sociální zabezpečení.....	26
2.3.4. Daň z příjmů FO ze závislé činnosti a funkčních požitků	26
3. Náhled současné situace v podniku.....	28
3.1. O firmě	28
3.2. Současná situace.....	28
3.3. Požadavky	29
3.4. Ostatní	30
4. Návrh databázové aplikace a implementace	31
4.1. Popis objektů reality	32
4.2. Tabulky a datový slovník	34
4.2.1. Tabulka tblOddeleni	34
4.2.2. Tabulka tblDodavatel.....	34

4.2.3. Tabulka tblZbozi.....	35
4.2.4. Tabulka tblObjednavka.....	35
4.2.5. Tabulka tblObjZbozi.....	36
4.2.6. Tabulka tblZamestnanec	37
4.2.7. Tabulka tblMzdoveUdaje	38
4.2.8. Tabulka tblVypocteneMzdy	39
4.3. Kardinalita vztahu mezi tabulkami	40
4.4. Aplikace	41
4.4.1. Zabezpečení	41
4.4.2. Formulář pro přihlášení	41
4.4.3. Formulář základního okna	42
4.4.4. Formuláře pro manipulaci se zbožím.....	43
4.4.5. Formuláře pro novou objednávku.....	45
4.4.6. Formuláře pro přehled a úpravu objednávek	47
4.4.7. Formuláře pro vytvoření a vypočtení nové mzdy	49
4.4.8. Formuláře pro přehled a úpravu mezd	50
4.4.9. Formuláře pro evidenci a správu zaměstnanců.....	52
5. Závěr.....	54
Seznam použité literatury.....	55
Seznam zkratk	57
Prohlášení o využití výsledků bakalářské práce	
Seznam příloh	

1. Úvod

Počítač? Pro lidstvo kdysi nepochopitelná a nedostupná věc, používána pouze pro vědecké disciplíny, popř. velké firmy. Dnes jsou počítače k dostání v jakémkoliv specializovaném obchodě a v různých formách (stolní počítače, notebooky, tablety, „chytré“ telefony....) se vyskytují prakticky ve všech firmách a dokonce téměř v každé domácnosti. Život bez nich je absolutně nemyslitelný. Se svým programovým vybavením ulehčuje život lidem a firmám napomáhá při analýze a při rozhodování v důležitých i méně důležitých podnikatelských situacích. Umožňuje nám skladovat a zpracovávat data v databázových systémech, čímž napomohl k částečnému odstranění papírových kartoték. V nejednom případě se v nich objevovaly stejné záznamy, tudíž docházelo k chybám a vznikal zmatek. Databáze nám přináší řád a jiný pohled na uchovávání důležitých dat.

Cílem bakalářské práce je navrhnout databázovou aplikaci pro malý podnik zabývající se nákupem a následným prodejem zboží koncovým zákazníkům. Účelem aplikace je zefektivnit mzdové a obchodní procesy ve firmě, které do dnešní doby byly zpracovávány buďto čistě papírovou formou, nebo nevyhovujícím softwarem. Po konzultaci s majiteli podniku byl stanoven závěr - aplikace bude kombinací objednávkového, evidenčního a mzdového systému. Objednávkový systém bude sloužit pro vytváření, správu a odesílání objednávek. Vedení jednotlivých dodavatelů a zaměstnanců firmy bude obsluhováno evidenčním systémem. Poslední část aplikace bude zaměřena na mzdový systém.

Bakalářská práce je strukturována do pěti hlavních částí.

V *Úvodu* je kompletně vyličená podstata bakalářské práce, její téma, kapitola teoretická i praktická.

V druhém oddílu *Přehled teoretických pojmů* budou čtenáři seznámeni s teorií, která bude v praktické části využita. Tento oddíl je ještě dále rozdělen do tří subkapitol *Návrh databáze, Microsoft Access a Mzdové účetnictví*. V návrhu databáze budou objasněny pojmy, jako jsou databáze, typy databází, relační databázový systém s jeho základními prvky tabulka, entita, atribut, doména, klíče a integrita. Nakonec budou představeny pojmy, jako je normalizace, třívrstvá architektura databázových systémů.

Práce bude vytvářena v programu Microsoft Access. Budou zde obeznámeny základní prvky aplikace – tabulky, formuláře a sestavy. Celá tato subkapitola bude završena jazykem SQL a Visual Basic for Application.

Jelikož databáze má některým zaměstnancům posloužit k výpočtu mezd ostatním, bude v subkapitole *Mzdové účetnictví* popsán princip počítání mezd. Budou rozebrány pojmy hrubé a čisté mzdy, zdravotní pojištění, pojistné na sociální zabezpečení, daň z příjmů fyzických osob, nemocenské a důchodové pojištění.

V *Náhledu současného stavu podniku* budou rozebrána fakta, jak to v maloobchodní jednotce skutečně chodí a jaké tam vznikají vztahy. V této podkapitole bude slovně charakterizován systém, který má být v rámci bakalářské práce navrhnut a bude upozorněno na všechny důležité i méně důležité prvky a vlastnosti, jenž databáze po konzultaci se zaměstnanci podniku musí mít.

Cílem předposlední kapitoly, *Návrh databázové aplikace a implementace*, budou ukázány, jaké tabulky byly vytvořeny a jaké jsou mezi nimi vztahy. Dále budou prezentovány vytvořené formuláře a tiskové sestavy, které budou jednotliví zaměstnanci používat.

V *Závěru* bude zhodnoceno provedení celé bakalářské práce, od zpracování teoretické části až po navrhnutou databázovou aplikaci. Budou uveřejněny i informace o reálném zavedení aplikace do provozu a zhodnocení, zda aplikace dopadla úspěšně či neúspěšně.

2. Přehled teoretických pojmů

Celá tato kapitola je věnována teoretické části bakalářské práce. Budou zde objasněny pojmy týkající se databází, prostředí programu Microsoft Access a problematiky mezd. Na základě této kapitoly bude vyhotovena praktická část práce.

2.1. Návrh databáze

Nejvýznamnější činností pro implementaci databázové aplikace do praxe je správně navrhnuté schéma databáze. Jaké tabulky bude obsahovat, s jakými atributy a datovými typy, v jakém vztahu budou jednotlivé tabulky korespondovat. Vše musí být naprojektováno tak, aby se v databázích zbytečně neobjevovala zdvojená data a aby tato data odpovídala realitě.

2.1.1. Databáze

Pod tímto často používaným pojmem je možné si představit obrovské množství dat pohromadě. Pro názornou představu, i bez světa počítačů, je možno zamířit do běžného života. Příkladem mohou být zdravotní karty, ve kterých jsou záznamy o jednotlivých pacientech. Tyto si doktoři vytvářejí, když je pacient poprvé navštíví a je jeho přáním ordinaci nadále navštěvovat. Karty jsou poté založeny k ostatním do kartotéky a pro rychlejší vyhledání jsou abecedně seřazeny. Dalším příkladem může být uchovávání faktur podle data vyhotovení v šanonech. Dnes se ale od výše zmíněného uchovávání dat pomalu ustupuje, a pokud to finanční prostředky dovolí, nechají si informační systém (IS) společnosti, podnikatelé, či firmy navrhnout. Příkladem využití databázového systému může být školní IS, který musí mít v databázi uložena data o studentech (jméno, příjmení, adresa, obor, program, atd.); elektronické obchody rovněž pracují s databází, ve které jsou uloženy veškeré údaje o nakupovaném výrobku a poslouží k vypočtení celkové ceny nákupu a mají mimo to mnoho dalších funkcí.

Data, informace

Veškeré získané údaje, jež jsou uloženy v tabulkách, představují data. Nemají žádnou strukturu, v čase se nemění do doby, pokud nejsou někým pozměněna. Celkově však mají důležitý význam buď pro jednotlivce, nebo pro organizaci. V momentě, když se data začnou zpracovávat (výstup ve formuláři, sestavě, atd.), stávají se z nich informace. Tímto zásahem dostávají skutečný smysl pro uživatele a mohou být neomezeně prezentována. Informace jsou měněny tak často, jak se mění v databázi. (Conolly, Begg a Holowczak, 2009; Hernandez, 2006)

Data jsou to, co ukládáme, informace jsou to, co získáváme. (Hernandez, 2006, s. 67)

Databázová aplikace vs. databázový systém

Na rozdíl od databáze, která se skládá čistě jenom z tabulek naplněnými daty, jsou součástí databázové aplikace formuláře, sestavy, s nimiž uživatel manipuluje pomocí požadavků, nejčastěji jazyka SQL¹. Pod pojmem databázový systém se vyskytuje veškerý SW (aplikace, databázový stroj², middleware³) i data. (Riordan, 2000)

Výhody databázového přístupu

- Je dohlíženo, aby se v databázi nevyskytovala *redundantní* (opakovaná) *data*.
- Odstraněním redundance je zajištěna *konzistence* (bezespornost) *dat* – hodnota datové položky bude vždy aktuální i po provedení změny, existuje-li pouze v jediné kopii.
- Jelikož je databáze přístupná všem pověřeným uživatelům, jsou *data* celé aplikace těmito uživateli *sdílena*.
- *Integrita* (neporušenost) *dat* souvisí s výše uvedeným pojmem konzistence dat. Je zabezpečeno, že data uložená v databázi odpovídají reálnému světu.
- Protože neexistuje vazba mezi popisem dat a aplikací, nemůže být aplikací pozměněn popis dat, jinak řečeno jsou *data nezávislá*. (Conolly, Begg a Holowczak, 2009)

2.1.2. Systém řízení báze dat

Systém řízení báze dat (SŘBD) představuje programové vybavení, které je součástí databáze. Jeho důležitou vlastností je, že představuje prostředníka mezi daty, aplikací a uživateli. Dává uživateli možnost práci s databází, a to *definovat ji, vytvářet a udržovat*. (Conolly, Begg a Holowczak, 2009)

SŘBD je nabízena celá řada dalších služeb. Uživateli je povoleno, aby data do databáze ukládal, měnil i mazal a následně je získával zpět (*Insert, Update, Delete, Select*). K datům je umožněn víceuživatelský přístup a je zabezpečeno, aby při aktualizaci různými uživateli ve stejném čase nedocházelo ke kolizi. Je zajištěno, že při provedení transakce⁴ budou do databáze zapsány všechny změny nebo žádná z nich. Podporuje práci pomocí dotazovacího jazyka SQL. Pro ochranu dat je použito zálohování a došlo-li by k nečekanému

¹ SQL – hlavní dotazovací jazyk pro relační databázi.

² Databázový stroj – mechanismy, které slouží pro ukládání dat na disk a opětovné načtení.

³ Middleware – program, pomocí něhož můžeme přistupovat z jedné databáze k druhé.

⁴ Transakce – skupina operací (příkazy, dotazy), které slouží pro práci s databází (přístup a modifikace).

selhání, je možné databázi obnovit. Obsahem jsou také různá bezpečnostní opatření, kterými je znemožněn přístup a modifikace dat neoprávněnými uživateli.

Nejčastěji používanými produkty SŘBD jsou *Microsoft Access*, *Microsoft SQL Server*, *Oracle Database*, ale často jsou používány i jiné produkty. (Oppel, 2008)

2.1.3. Třívrstvá architektura databázových systémů

Účelem třívrstvé architektury databázových systémů je rozlišování mezi uživatelskou aplikací a fyzickou databází. Z tohoto důvodu je na databázi pohlíženo ve třech úrovních: *externí, konceptuální a interní*, jak je možné vidět na obrázku 1.

a) Externí úroveň (External Level)

Externí úroveň je založena na tom, že každému uživateli bude zobrazen individuální pohled⁵, podle toho s jakými daty potřebuje pracovat, nebo jaká data potřebuje získat, či prezentovat. Ostatní *atributy, entity a relace* v databázi budou stále existovat, aniž by uživatel tušil, že by v dané databázi mohly být uloženy, všímá si pouze těch svých. (Conolly, Begg a Holowczak, 2009)

b) Konceptuální úroveň (Conceptual Level)

Konceptuální úroveň je představována pomyslnou hranicí mezi externí a interní úrovní, čímž je umožněna *logická⁶ a fyzická⁷ nezávislost dat*. Typické pro tuto úroveň je, že charakterizuje, jaká data se v databázi objevují a jaké vztahy se mezi nimi vyskytují. Právě zde jsou uloženy všechny *atributy, entity a relace*, ze kterých poté konkrétní uživatel čerpá ty pro svou potřebu. Neměla by být opomenuta ani integrita, jež s daty taktéž úzce souvisí. A rovněž by neměla být přehlédnuta ani *sémantická metadata⁸* a *bezpečnostní informace*. (Conolly, Begg a Holowczak, 2009)

c) Interní úroveň (Internal Level)

V interní úrovni je objasněn fakt, jak jsou data uložena v databázi a jak jsou vnímána relačním databázovým systémem (DBMS) a operačním systémem (OS). Touto úrovní je přispěno k nepříznivějšímu *běhovému výkonu a využití prostoru při uložení*. Aby bylo zcela

⁵ Pohled – vytvoření pomyslné tabulky na popud uživatelského dotazu, který z původně vytvořených tabulek zobrazí pouze to, co uživatel potřebuje ke své práci.

⁶ Logická nezávislost dat – dojde-li ke změně v konceptuálním schématu, tato změna se neprojeví v externích schématech.

⁷ Fyzická nezávislost dat – dojde-li ke změně v interním schématu, tato změna se neprojeví v konceptuálním schématu.

⁸ Metadata – informace o datech

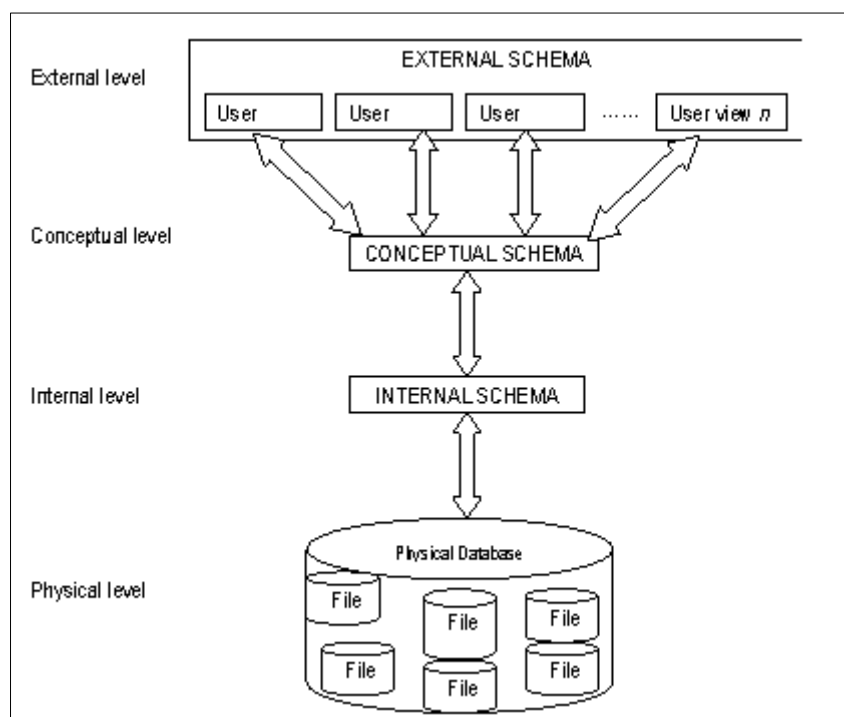
jasné, jak mají být data na paměťových nosičích uložena, jsou pro tuto část navrženy struktury datové a struktury souborů. (Conolly, Begg a Holowczak, 2009)

Další nedílnou součástí, pro korektní fungování databáze, je vytvoření rozhraní pro metody, které obsluhují OS:

- při ukládání dat na paměťová zařízení,
- vytváření indexů,
- přístupu k datům, atd. (Conolly, Begg a Holowczak, 2009, s. 52)

Výhody třívrstvé architektury

- Všichni uživatelé by měli mít k dispozici stejná data. Pohledy jsou přizpůsobeny jejich potřebám. Pokud uživatel pohled změní, nemá to žádný vliv na ostatní účastníky.
- Uživatelé se nezabývají *fyzickým uložením dat*. Jeho práce s databází nemá nic společného s organizací *fyzického uložení*.
- Uspořádání *fyzického uložení* by nemělo být závislé na interní struktuře databáze.
- Pozměnění-li správce databáze (DBA), který má tuto úroveň na starosti, uložení, či konceptuální stavbu databáze, nemělo by to mít žádný vliv na uživatele. (Singh, 2009)



Obrázek 1.: Třívrstvá architektura databázových systémů. (Singh, 2009)

2.1.4. Relační databáze

Důležitou osobou v oblasti relačních modelů je E. F. Codd z firmy IBM, jenž v roce 1970 relační model prezentoval jako první. Architektura relační databáze je založena na základech matematiky, konkrétně *teorie množin a predikátové logiky*. Od ostatních síťových a hierarchických modelů se liší tím, že se nemusí starat o fyzické uložení dat. V 70. letech představovala nezávislost reprezentace fyzického uložení dat obrovský průlom, dnes už je toto považováno za samozřejmost. (Riordan, 2000)

Základnu relační databáze tvoří *relační tabulka* (relace⁹), avšak konečnou databází je myšlena celá řada tabulek obsahujících data, která mají vzájemnou souvislost a jsou mezi sebou propojena. Jako každou jinou dvourozměrnou tabulku ji reprezentuje řádek a sloupec. Důležité je v relační databázi navrhnout *datový model*, jenž představuje obecné řešení problému. Datový model se skládá z *entit, atributů, domén a vztahů*. S touto problematikou též souvisí *primární klíč, cizí klíč a funkční závislost*. Velmi důležitý pojem, spojen s tématem relačních databází, je *integrita dat*. (Skřivan [online], 2002; Riordan, 2000) Všechny zmíněné komponenty relační databáze budou podrobně rozebrány o kapitole níž.

Relační typ databáze je mezi ostatními databázemi ze všech nejpoužívanější i nejoblíbenější, a to hlavně pro svou jednoduchost, pochopitelnost a pro možnost implementace i do složitějších organizačních struktur. Relační databáze jsou nejčastěji používány k inventarizaci majetku, správě pacientů, bankovníctví, zpracování objednávek, plánování událostí a rezervaci letenek. (Skřivan [online], 2002; Hernandez, 2006)

2.1.5. Komponenty relační databáze

Mezi základní komponenty relačních databází jsou řazeny relační tabulky, klíče, vztahy a integrita.

Relační tabulka

Relační tabulka je reprezentována řádky a sloupci jako každá jiná dvourozměrná tabulka.

Při návrhu databáze jsou veškeré tabulky nazývány slovem entity. Tento termín je označení pro *událost, osobu, místo, věc* (např. knihy, objednávky, zaměstnanci), o které se

⁹ Relace – je součástí matematické disciplíny teorie množin. Jeho pojem může pokrývat dva názvy, a to tabulku nebo vztah.

v tabulce uchovávají data. Řádky v navrhované tabulce jsou prezentovány n-ticí záznamů, čili instanci konkrétní události, osoby, místa či věci (jeden řádek = jeden záznam).

Na druhou stranu sloupce jsou myšleny atributy, jimiž je charakterizována každá jednotlivá buňka (neboli pole) dané entity. V tabulce se kromě uložených záznamů vyskytuje zvláštní první řádek, který je označen jako hlavička. Tou jsou popsány atributy a je jí dán význam, jaká budou do daného sloupce zapsána data. Atribut by měl obsahovat pouze jednu hodnotu a stejný název atributu v jiné tabulce by měl splňovat totožný význam, aby nedocházelo k omylu. Do polí by se neměly ukládat jakékoli výpočty, např. spojování textových řetězců či matematické propočty. Až výsledkem vytvořeného dotazu se výpočty mohou objevovat ve virtuální tabulce. (Hernandez, 2006) Příklad klasické tabulky můžeme vidět na obrázku 2.

Všechny tabulky musí obsahovat identifikační atribut, který bude odlišovat každý jednotlivý záznam, tím bude typický a rozdílný od ostatních. Tomuto identifikačnímu atributu se říká klíč, jenž bude podrobněji analyzován v další části. Klíč může být složen z kódu (např. číslo objednávky, osobní číslo studenta, zaměstnance) a v případě, že kód není možné z něho sestavit, je použit datový typ automatické číslo. Podle Oppla (2008) zde musí být bráno na vědomí, že automatické číslo ani identifikační kódy nemají vliv na fyzické uložení dat v paměti.

ID ▾	Jmeno ▾	Prijmeni ▾	Ulice ▾	CisloPopisn ▾	Mesto ▾	Stat ▾
1	Jan	Novák	Chrudimská	25	Jindřichův Hradec	Česká republika
2	Marie	Sikorová	Dlouhá	854	Zlín	Česká republika
3	Jindřiška	Turková	Malostranská	96	Praha	Česká republika

Obrázek 2.: Ukázka tabulky (Vlastní práce)

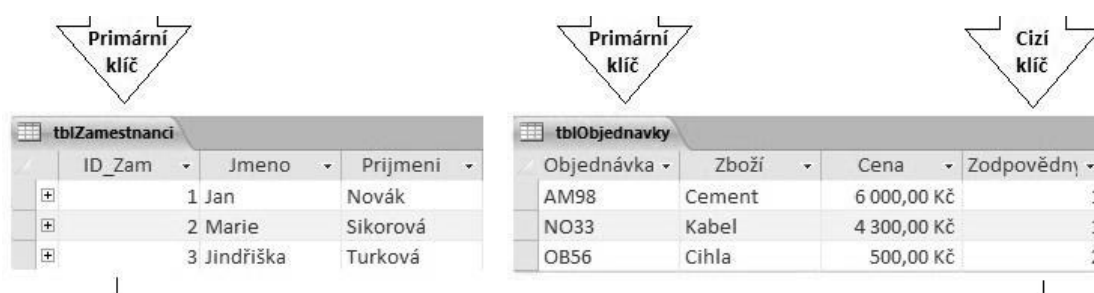
Klíče

Už bylo zmíněno, že tabulka musí ve své struktuře obsahovat identifikační atribut, jinak řečeno *klíč*. Klíč musí být v rámci celé databáze jednoznačný, nemůže existovat více klíčů se stejným názvem a každý z nich měl jiný význam.

V relačních databázích se rozlišuje mezi dvěma druhy klíčů, primární a cizí klíč. Primární klíč je mnohem důležitější než cizí klíč. Primární klíč jednoznačně odlišuje jednotlivé záznamy, což znamená, že každá hodnota záznamů v primárním klíči je různá. Může se skládat z jednoho atributu, nebo z více atributů, potom je takový klíč označován jako *složený primární klíč*.

Cizí klíč je vytvořen tak, že je zkopírován primární klíč z tabulky, která ho ve svém těle zahrnuje, do tabulky s kterou má logickou souvislost. Označení cizí klíč je proto, že tabulka, ve které je vložena kopie primárního klíče, už svůj primární klíč také má. Z toho vyplývá, že tabulka se může skládat jak z primárního klíče, tak i z cizího. (Hernandez, 2006) Na obrázku 3 je znázorněno, že tabulka tblZamestnanci je složena pouze z primárního klíče, kdežto tabulka tblObjednavky už má oba klíče – svůj primární klíč a kopii primárního klíče z tabulky tblZamestnanci (cizí klíč).

Tyto klíče jsou určeny k tomu, aby mohla být vytvořena vazba mezi tabulkami, které spolu svým způsobem souvisí.



Obrázek 3.: Ukázka primárních a cizích klíčů mezi dvěma tabulkami (Vlastní práce)

Vztahy

Vztah symbolizuje pojem, pomocí něhož jsou znázorněny vzájemné souvislosti mezi jednotlivými tabulkami. Vztah je velice důležitá komponenta relační databáze, protože položeným dotazem je požadováno získávat různá data z různých tabulek, které mezi sebou mají určitou logickou vazbu. (Oppel, 2008) Vztahy také napomáhají k tomu, aby se v tabulkách co nejméně objevovala redundantní data. (Hernandez, 2006)

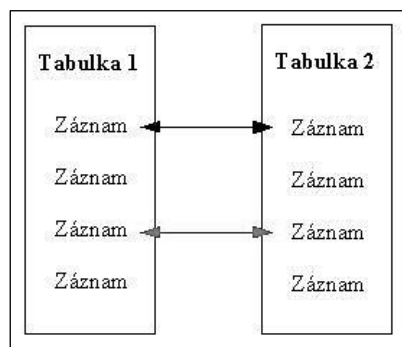
Pro představu je vytvořena tabulka tblObjednavky se svými atributy a samozřejmě s alespoň jedním záznamem. Nyní postačuje hlavně atribut, kód zaměstnance, jenž bude představovat osobu, která je za danou vyhotovenou objednávkou zodpovědná. Dále je zhotovena tabulka tblZamestnanci, ve které je obsažen jak kód zaměstnance, tak i další charakteristické údaje. Aby v tabulce tblObjednavky nemusely být zaznamenány veškeré atributy o konkrétním zaměstnanci, nýbrž jen kód zaměstnance, existují v relačních databázích výše zmíněné vztahy. V případě že bude nutné zjistit, kdo je za objednávku zodpovědný, budou díky této komponentě zpřístupněny všechny údaje z tabulky tblZamestnanci.

Na konceptuální úrovni jsou rozlišeny tři typy kardinalit vztahu (1:1, 1:N a M:N).

a) 1 : 1

Typ vztahu 1:1 je vyznačován tím, že záznam v první tabulce (nadřazená tabulka) odpovídá právě jednomu záznamu ve druhé tabulce (podřazená tabulka). Rovněž záznam ve druhé tabulce má pouze jeden protějšek v první tabulce, jak je vidět na obrázku 4. Tento vztah se vyskytuje velice zřídka. Používá se zejména v případech, kdy existuje tabulka s mnoha atributy a pro přehlednost je rozdělena do dvou tabulek.

Je-li potřebné k nadřazenému záznamu zřídčit související záznam v podřazené tabulce, musí již tento záznam v nadřazené tabulce existovat. Vztahy jsou vytvářeny tak, že kopie primárního klíče nadřazené tabulky je vložena do podřazené tabulky, kde už plní svou funkci cizího klíče. V případě tohoto vztahu může cizí klíč podřazené tabulky dále sloužit jako klíč primární. (Hernandez, 2006)



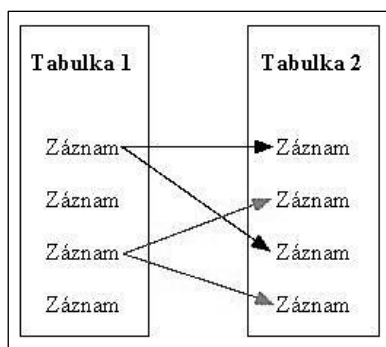
Obrázek 4.: Ukázka vztahu 1:1 (Hauzar [online], 2003)

b) 1 : N

Vztah 1:N patří mezi nejčastěji aplikované vztahy. Teorie spočívá v tom, že z jednoho záznamu v první tabulce je možné se odkázat na jeden či více záznamů ve druhé tabulce, ale opačně již toto možné není. Z druhé tabulky je možnost se odkázat pouze na jeden jediný záznam v první tabulce. Tento vztah je znázorněn na obrázku 5.

Příkladem může být vztah mezi zaměstnanci a zaměstnavatelem, kdy pod jednoho zaměstnavatele spadá několik zaměstnanců, ale zaměstnanci patří pouze jednomu zaměstnavateli.

Vytvoření tohoto typu vztahu je celkem srovnatelné se vztahem 1:1. Primární klíč z první tabulky je zkopírován do druhé tabulky, kde jako v předcházejícím případě je určen jako cizí klíč. (Hernandez, 2006)



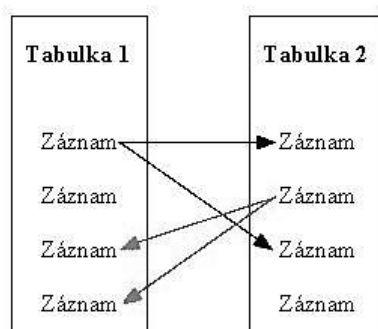
Obrázek 5.: Ukázka vztahu 1:N (Hauzar [online], 2003)

c) M : N

Mezi poměrně složitější vztah, co se vytvoření týče, patří vztah M:N. Tento vztah vznikne, když záznam v první tabulce odpovídá jednomu, nebo několika záznamům ve druhé tabulce. Zároveň platí i opačný proces, kdy jeden záznam ve druhé tabulce odpovídá jednomu nebo několika záznamům v první tabulce. Jak je to ve skutečnosti, je vidět na obrázku 6.

Typickým příkladem tohoto vztahu může být objednávka vs. zboží. Tedy objednávka může obsahovat několik různých záznamů zboží a na druhé straně zboží může být součástí několika různých objednávek.

Jak už bylo řečeno, je u vztahu M:N problém s jeho vytvořením. Není jasné daný způsob, jak mají být související tabulky propojeny. Zda mají být připojeny záznamy z jedné tabulky do druhé, nebo ty z druhé do první. K tomuto problému je vymezena *vazební tabulka*. Ta slouží jako spojovatel mezi dvěma tabulkami, který umožní propojení jejich dat. Nová tabulka je vytvořena tak, že je do ní zkopírován primární klíč z Tabulky 1 (M) a z Tabulky 2 (N). (Hernandez, 2006)



Obrázek 6.: Ukázka vztahu M:N (Hauzar [online], 2003)

Integrita

Integritou by mělo být zabezpečeno, aby data v databázi byla přesná. Pokud nebudou dodržena striktní pravidla, která s integritou souvisí, budou v databázi vznikat chyby, jež bude v zaběhnuté databázi těžké najít a odstranit. Proto je velice důležité přípravu nepodcenit. V potaz musí být brány všechny čtyři druhy integrit – *relační integrita*, *entitní integrita*, *doménová integrita* a *business pravidla*.

Referenční integrita je spojena s cizím klíčem. Cizí klíč by měl zahrnovat takové údaje, které jsou obsaženy v primárním klíči, z něhož byla vytvořena kopie, případně by neměl obsahovat žádný údaj. Referenční integritou jsou synchronizovány údaje mezi tabulkami, které mají mezi sebou vazbu. Použití je při ukládání, mazání a získávání záznamů

Entitní integrita je spjata s pojmem primárního klíče, kde je nutné, aby byl atribut jedinečný v rámci celé databáze. Dále by mělo být zajištěno, aby se v databázi nevyskytovaly zdvojené záznamy. Posledním pravidlem je, že hodnoty primárních klíčů nesmí být null¹⁰.

Doménová integrita je zaměřena na ostatní hodnoty atributů a jejím cílem je dosáhnout, aby tabulky ukládaly pouze spolehlivé hodnoty. Potřebujeme pracovat s daty, která jsou *platná, konzistentní a přesná*.

Výše zmíněné integrity jsou zavedeny na úrovních *tabulek, polí a vazeb*, ale je potřebné zavést také omezení při získávání a práci s daty. Toto omezení je pojmenováno jako business pravidla. Pravidla jsou sestavena podle konkrétního fungování a vztahů v podniku. Je nezbytné vědět, jaká data mají být uložena, jaká získána. Kdo všechno ke své práci může využívat databázi a v jakém rozsahu, jestli budou použity bezpečnostní prvky nebo ne, atd. (Conolly, Begg a Holowczak, 2009; Hernandez 2006)

2.1.6. Normalizace

Normalizace byla vytvořena, aby se v rámci databázových systémů zredukovalo co největší množství redundancí. Obsahují-li tabulky redundantní data, mohou vznikat problémy při aktualizaci (vkládání, mazání a úpravě). Proto je důležité, aby tabulky byly v normální formě. Podle Riordan (2009) existuje několik normálních forem: první normální forma (1NF), druhá normální forma (2NF), třetí normální forma (3NF), Boyce/Coddova normální forma (BCNF), čtvrtá normální forma (4NF), pátá normální forma (5NF). K odstranění redundantních dat je postačující použít 3NF. První tři normální formy budou v práci popsány.

¹⁰ Null – tímto výrazem je myšlena pouze prázdná buňka, nikoliv nula, či prázdný řetězec (mezera).

Jelikož normalizací je relace rozdělena na menší části, kterými jsou představovány jednotlivé relace bez redundancí, musí být zajištěna bezztrátová dekompozice, kdy spojením relací nesmí dojít k žádné ztrátě.

První normální forma

V rámci relačních databází je první forma nejdůležitější a znamená, že atribut je tvořen pouze jednou hodnotou (čili dále nedělitelný). Z praktického hlediska však mohou vzniknout nesrovnalosti. Záleží na jedinci, jaké bude v databázi uchovávat data a k čemu budou sloužit, jestli např. adresu bude považovat za konečnou nebo ji bude muset rozdělit (město, ulice, č. p., PSČ).

Druhá normální forma

Druhá normální forma je splněna, pokud je relace v 1NF a sloupce, které netvoří primární klíč, přímo závislé na primární klíči dané tabulky.

Tato problematika se týká pouze relací, které obsahují složený primární klíč. Má-li relace pouze jeden primární klíč, bude vždy v 2NF.

Třetí normální forma

Relace je v třetí normální formě, pokud je v 2NF a neexistuje závislost mezi atributy, které netvoří primární klíč. V praxi je 3NF představována nejvyšším stupněm normalizace, kdy je 3NF zabezpečeno, že relace nebude obsahovat redundance.

2.1.7. Bezpečnost

Při vytváření databáze musí být jednoznačně rozhodnuto, jestli jsou uchovávaná data důležitá. Zda musí být použita ochranná opatření a nastavena přístupová práva, anebo ne. Budou-li do databáze ukládána data, která pro firmu nejsou citlivá a jejím případným únikem nepoškodí fungování společnosti, nemusí být žádná bezpečnostní politika při implementaci do podniku zavedena. Většinou však nastává situace, kdy data musí být chráněná a ne každý má přístup pro práci s databází povolený, případně má uživatel přístup pouze omezený. Následně se podle Riordan (2000) rozlišuje mezi *bezpečností na sdílené úrovni*, kdy je všem přiřazeno stejné heslo, nebo *uživatelskou bezpečností*, kde má každý uživatel své heslo a práva.

2.2. Microsoft Access

Microsoft Access je aplikace, která slouží k vytváření a správě databází. Access slouží pro návrh tabulek, nadefinování jejich vztahů a zhotovení konkrétních formulářů, či sestav pro jednodušší a přehlednější manipulaci s daty. K vytváření dotazů je možné přímo použít jazyk SQL (Structured Query Language) a pro laiky, kteří tento jazyk neovládají, je v Accessu speciální návrhové zobrazení. Prvkům formulářů lze naprogramovat různé funkce s využitím programovacího jazyka VBA (Visual Basic for Application).

2.2.1. Tabulka

Jak vypadá klasická tabulka a z čeho se skládá, už bylo probráno. Nyní budou popsána různá zobrazení, které je možné v Accessu využít a co všechno v nich lze nastavit. Mezi nejpoužívanější zobrazení je řazeno *Návrhové zobrazení*, kde jsou vytvářeny jednotlivé sloupce s jejich datovými typy a případně doplňujícími vlastnostmi. Dále *Zobrazení Datový list*, kde jsou v řádcích zapisovány hodnoty přednastavených sloupců. Rovněž je zde možnost přidat nový sloupec, ale již bez vlastností. A méně používanými zobrazeními jsou *Zobrazení kontingenční tabulky* a *Zobrazení kontingenčního grafu*.

2.2.2. Datové typy a vlastnosti

Každé vyhotovené tabulce odpovídá konečná množina sloupců, kde každému zvlášť je přiřazen datový typ a jsou nastaveny vlastnosti.

Datové typy

Při vytváření jednotlivých sloupců je možné vybírat z několika datových typů.

- *Text* – pole může být vyplněno textovými i číselnými hodnotami.
- *Memo* – pole je vyplněno dlouhým textem i s případným formátováním.
- *Číslo* – do pole tohoto typu budou ukládány číselné hodnoty.
- *Datum a čas* – slouží pro ukládání časových hodnot.
- *Měna* – pokud se ví, že bude potřeba použít peněžních jednotek.
- *Automatické číslo* – využívá se k identifikaci a odlišení jednotlivých hodnot, použití u primárních klíčů.
- *Ano/ne* – pokud ke specifikaci pole postačí logická hodnota.
- *Objekt OLE* – tímto datovým typem je přidán objekt OLE (např. tabulka z Excelu).
- *Hypertextový odkaz* – je určen pro uložení e-mailové adresy nebo URL adresy webové stránky.

- *Príloha* – má podobný princíp jako objekt OLE, jen je podporován větším množstvím formátů.
- *Průvodce vyhledáváním* – je vytvořen seznam (roletka) a z nich bude některá vybrána. (JaknaOffice.cz [online], 2011)

Vlastnosti

Ke každému jednotlivému datovému typu lze nastavit mnoho různých vlastností, hodně jich je společných a některé jsou specifické pro daný datový typ.

- *Velikost pole* – nastavení maximální velikosti pole.
- *Formát* – nastavení vzhledu řetězce – např. měna, procento (zobrazení v datovém listu, tiskové sestavy, formuláře, atd.).
- *Vstupní maska* – představuje nastavení přesně daného vzhledu, v jakém musí být text, číslo nebo měna uložena. Maska je vytvářena pomocí různých znaků, např. 0 – povinná číslice, 9 – nepovinná číslice, L – povinné písmeno, ? – nepovinné písmeno,
- *Titulek* – pokud bude titulek vyplněn, bude v různých zobrazeních použit jako název pole.
- *Výchozí hodnota* – při přidávání nového záznamu se nám tato hodnota automaticky zobrazí, ale je ji možné přepsat.
- *Ověřovací pravidlo* – do pole musí být zadán takový záznam, který bude odpovídat vytvořenému pravidlu.
- *Ověřovací text* – souvisí s ověřovacím pravidlem, pokud záznam nebude odpovídat pravidlu, bude zobrazen tento ověřovací text.
- *Je nutno zadat* (Ano/ne) – pomocí této vlastnosti je určeno, zda je vyplnění záznamu povinné.
- *Povolit nulovou délku* (Ano/ne) – je stanoveno, jestli záznam může obsahovat nulový řetězec (= prázdnou hodnotu).
- *Indexovat* (Ano - s duplicitou / bez duplicity / ne) – indexem je dosaženo rychlejší vyhledávání. S duplicitou mohou mít záznamy stejnou hodnotu, bez duplicity nikoli. (Písek, 2011)

2.2.3. Formuláře

Formulářem se rozumí grafické rozhraní¹¹, jehož cílem je, aby uživatelům usnadnil a zpřehlednil manipulaci se záznamy. Do formuláře mohou být zahrnuta data z více tabulek, které mezi sebou mají vazbu. Data ve formulářích mohou být vyhledána, potom upravena či smazána, nebo vložena nová data. Uložení provedené změny se projeví vždy ve zdrojové tabulce, nikdy ne přímo ve formuláři. Formulářem jsou data z tabulek pouze zobrazována a při provedené změně musí být formulář obnoven.

Formulář lze vytvořit několika způsoby, jednodušší a méně pracný je automaticky vygenerovaný formulář tlačítkem *Formulář*. Složitější způsob, ale šitý na míru potřebám, je zhotoven v rámci *Prázdný formulář*, kde se vše navrhuje od začátku a v nově otevřeném formuláři se nevyskytují žádné ovládací prvky. Ty jsou vkládány až návrhářem, podle toho k čemu má formulář sloužit. Ke zjednodušení se může použít *Průvodce formulářem*, kde je přednastavenými kroky dosaženo daného vzhledu.

Zjednodušení formuláře spočívá v tom, že obsahuje ovládací prvky, jimiž je zapříčiněna přehlednost ve formuláři. Dále je ve formulářích poskytnuto několik navigačních prvků, pomocí kterých se ve formuláři pohybuje a orientuje. Mezi ovládací prvky se řadí např. *Textové pole*, *Popisek*, *Tlačítko*, *Pole se seznamem*, *Graf* a mnoho dalších. Také jako tabulka má i formulář několik možných zobrazení, většina z nich se neliší. Navíc je tam *Formulářové zobrazení*, kterým je prezentováno, jak vytvořený formulář se záznamy skutečně vypadá, ale již v něm nelze upravovat. *Zobrazení rozložení*, kde je vidět vzhled formuláře i se záznamy. V tomto módu je možné s ovládacími prvky ještě pohybovat a upravovat je.

2.2.4. Sestavy

Sestavy neboli tiskové sestavy, jsou určeny výhradně pro tisk. Jejich cílem je poskytnout uživateli přehlednější strukturu výtisku. Nelze v nich žádným způsobem data upravovat. Stejně jako formuláře se mohou záznamy skládat z různých zdrojů, které mají mezi sebou vztah.

Způsob, jakým lze sestavu vytvořit, se příliš neliší od vytváření formulářů. Na výběr je opět automatická sestava vygenerována tlačítkem *Sestava* nebo může být použita *Prázdná sestava*, kde je vše vyhotoveno dle vlastních přání. Navrhování může být opět zjednodušeno využitím průvodce - *Průvodce sestavou*, ve kterém je procházením dílčích kroků dosaženo

¹¹ Grafické rozhraní – pomocí oken a různých ovládacích prvků (tlačítka, ikony) usnadňuje uživatelům práci s aplikací.

výsledku. Při návrhu sestavy je možné pracovat ve třech oblastech - podrobnosti, pod čímž je myšleno tělo sestavy, záhlaví a zápatí. Záhlaví a zápatí se rozděluje do dvou úrovní. Na úrovni stránky, kdy záhlaví a zápatí bude zobrazeno na každé stránce. Na úrovni sestavy, kdy bude zobrazeno na začátku a na konci sestavy. V rámci sestav je umožněno provádět různá řazení podle vybraných záznamů i jakékoli agregace, což u formulářů neexistuje. Pokud je vytvořena určitá skupina dat, má tato skupina v návrhu sestavy k dispozici své podrobnosti.

2.2.5. Výběr dat

Někdy uživatel potřebuje ke své činnosti pouze určitou skupinou dat, např. chce, aby se mu zobrazili zaměstnanci starší 60 let. K tomu aby takové zaměstnance bylo možné vybrat, lze použít filtry nebo dotazy, příp. jazyk SQL.

Filtry

Filtry jsou vytyčeny k rychlému, jednoduchému a omezenému zobrazení dat z databáze podle kritérií, které si uživatel sám zadá. Data budou vygenerována podle toho, jak je potřeba. Použitím filtrů se neprovádí žádná změna ve struktuře uložených dat v tabulce ani ve vytvořeném formuláři, ani v sestavě. Pokud je filtr zrušen, objeví se původní tabulka, formulář či sestava. Filtry lze provádět pouze nad jednou tabulkou, jedním formulářem, nebo jednou sestavou.

Rozlišuje se mezi čtyřmi typy filtrů. *Běžné filtry*, *Filtry podle výběru*, *Filtry podle formulářů*, *Rozšířené filtry*. Běžné filtry jsou v aplikaci už předem nadefinované a stačí je pouze kliknutím vybrat. Filtry podle výběru souvisí s polem v tabulce, ve kterém se pohybuje aktivní kurzor. Podle této hodnoty je možné filtrovat. Ve filtrech podle formulářů je vytvořen nový fiktivní formulář, nebo datový list skládající se z hodnot, které odpovídají kritériu. V rozšířených filtrech je vše psáno ručně, protože je možné filtrovat podle potřeby. (Microsoft, 2014)

Dotazy

Dotazy jsou vytvářeny za účelem výběru jen některých sloupců z jedné, nebo více tabulek, které jsou spolu ve vztahu. Možné je nastavovat různá kritéria, jež zobrazí pouze hodnoty odpovídající danému kritériu (výběrové dotazy). V podstatě řeší stejný problém jako filtr. Od filtru se dotaz liší tím, že je uložen jako nový objekt databázové aplikace a je možné ho kdykoli opět použít. S daty vybranými dotazem je možné provádět různé akce - výběr lze smazat, aktualizovat nebo převést do nové tabulky (akční dotazy). Dotazem lze také dopočítat

nové pole s novými hodnotami. Souhrnem jsou seskupeny stejné hodnoty. Těmto hodnotám je možné přiřadit určitou funkci (suma, součin, max/min, atd.).

Dotaz je možno vyhotovit pomocí *Průvodce dotazem*, *Návrh dotazu* nebo jazykem SQL. Při vytváření dotazu se vyplňuje *Pole* (sloupec tabulky), *Tabulka* (zdrojová tabulka), *Řadit* (vzestupně/sestupně), *Zobrazit* (má-li být pole viditelné nebo ne), *Kritéria* (viz níže; v jazyce SQL klauzule where), příp. *nebo*; je-li použit souhrn, je navíc zobrazeno *Seskupit*. V kritériích je velmi důležité znát všechny operátory. Mimo ty nejznámější (<, >, <=, >=, ><) zde patří:

- **In**(hod1, hod2,...) – bude zjištěno, zda se hodnoty vyskytují ve výčtu.
- **Between** hod1 **And** hod2 – budou vybrány hodnoty mezi hod1 a hod2.
- **Is Null** / **Is Not Null** – hodnota v poli je nulová/nenulová, neobsahuje žádný/ obsahuje jakýkoli řetězec.
- **Like** / **Not Like** – hodnota musí/nesmí obsahovat daný řetězec. Tento operátor využívá tyto metaznaky:
 - ? / * – představuje zastoupení pro jeden/neomezený počet znaků.
 - # – představuje zastoupení pro jednu číslici.
- **AND** / **OR** – a zároveň/anebo. (Písek, 2010)

Jelikož existuje několik datových typů, jsou různě odlišeny: text musí být v uvozovkách, datum/čas v hashích (#) a číslo není nutné diferencovat.

Vyhotovené dotazy v grafickém rozhraní Accessu jsou automaticky konvertovány do jazyka SQL. Není však zakázáno se přepnout do návrhu *Zobrazení SQL* a napsat vše ručně.

Structured Query Language (SQL)

Jazyk SQL je určen pro relační databáze a je podporován skoro všemi SŘBD. Prostřednictvím SQL jsou psány dotazy¹². SQL je jazyk neprocedurální, kdy již není nutné, aby uživatel programoval funkce, jež existují. Stačí znát pouze název funkce, kterou chce použít.

V Microsoft Accessu je nejčastěji použito klíčové slovo SELECT z jazyka *Data Manipulation Language (DML)*, kde mimo jiné patří *INSERT*, *UPDATE* a *DELETE*. Klíčové slovo SELECT je určen k výběru, jaká data mají být zobrazena (název sloupce, konstanta, výraz SQL; výběr všech sloupců - *; shodné záznamy se nebudou/budou opakovat – *SELECT*

¹² Dotaz – je požadavek, který je poslán databázi a uživateli je vrácena odpověď.

DISTINCT/ALL). Při psaní konkrétního dotazu je v psaní dotazu striktně dodržováno pořadí jednotlivých klauzulí, podle Nováka (2012) *SELECT, FROM, WHERE, ORDER BY, GROUP BY, HAVING*.

Klauzulí *FROM* je vytyčena tabulka, ze které jsou data získána. Zde může dojít i ke spojení tabulek:

- kartézský součin – záznamy ze všech tabulek,
- *INNER JOIN* – vnitřní spojení – spojení při shodě záznamu,
- *LEFT/RIGHT OUTER JOIN* – vnější spojení – z jedné z tabulek budou zobrazeny všechny záznamy.

Klauzule *WHERE* úzce souvisí s kritérii při vytváření dotazu. Budou zobrazena jen ty záznamy, které odpovídají dané podmínce. Pomocí klauzule *ORDER BY* jsou záznamy řazeny podle zadaného názvu sloupce, a to sestupně (*DESC*) nebo vzestupně (*ASC*). Klíčové slovo *GROUP BY* odpovídá souhrnným dotazům a plní stejnou funkci. Pokud v souhrnech bude nutné vybrat seskupená data podle nějakého kritéria, bude použita klauzule *HAVING*. Pokud nejsou použity souhrny, klauzule *HAVING* se v SQL příkazu nevyskytuje.

2.2.6. Visual Basic for Application (VBA)

VBA je programovací jazyk navrhnut firmou Microsoft pro MS Office (Excel, Access) a působí nad OS Windows. Je řazen do tzv. interpretovaných jazyků, protože napsaný kód je prováděn instrukci po instrukci.

S VBA souvisí pojem modul, což je kolekce deklarací, příkazů a procedur. Podle Nováka (2012) se moduly rozdělují do dvou kategorií: modul třídy a standardní modul. Modul třídy (neboli modul formuláře či sestavy) je svázán s konkrétním formulářem či sestavou. Čili je vymezen k naprogramování prvků formuláře či sestavy. Např.: co se provede při kliknutí na tlačítko, stisknutí tlačítka myši, přjetím přes určitý objekt formuláře. Na druhé straně standardní modul je tvořen procedurami, které jsou dostupné v rámci celé aplikace.

S moduly také souvisí proměnné a jejich doba platnosti. Proměnnou je vymezen určitý prostor v paměti, do kterého bude uložena výsledná hodnota. Tato je uložena pouze po dobu, kdy je program v běhu. Velikost rezervované paměti je odvozena od zvoleného datového typu. Dostupnost a platnost proměnné závisí na tom, v jakém modulu se nachází a jakou má deklaraci. Nachází-li se v proceduře, je k dispozici pouze v rámci dané procedury.

Je-li v modulu třídy, je dostupná pouze v daném objektu (modul formuláře či sestavy). Jedině ve standardním modulu ji lze využít v celé aplikaci.

Napsaný kód v jazyce Visual Basic for Application, kterým je zapříčiněná požadovaná akce, se nazývá procedura. Procedura je složena z příkazů a metod. Rozlišuje se mezi dvěma typy procedur. Jedna je charakteristická navrácením hodnoty (*Function*). Druhá pouze provede napsaný kód a nevrací žádnou hodnotu (*Sub*). (Shepherd, 2012)

2.3. Mzdové účetnictví

V této podkapitole bude řešena základní oblast týkající se mezd.

2.3.1. Hrubá a čistá mzda

V mzdovém účetnictví se rozlišuje mezi pojmy mzda a plat. Mzdou je myšlena peněžní částka za provedenou práci, která je poskytována zaměstnavatelem v soukromém sektoru. Pojem plat je používán v případě, že je zaměstnavatelem stát, územní samosprávný celek, státního fond, příspěvková organizace nebo školská právnická osoba.

Mzda je složena z pevné a pohyblivé částky. Pohyblivá část se odvíjí od výkonu zaměstnance a ochoty zaměstnavatele ji vyplatit. Může se jednat o odměny, příplatky, osobní ohodnocení a prémie. Odměna je ve většině případů vyplácena jednorázově, např. za kvalitně provedenou práci. Za příplatky se považuje práce přesčas, ve svátek, noční směny, práce ve ztíženém pracovním prostředí a práce v sobotu a v neděli. Pokud zaměstnanec vykazuje dlouhodobě kvalitní práci, může mu být poskytnuto osobní ohodnocení. Prémie jsou zaměstnavatelem uděleny, aby zaměstnancům byla zvýšena motivace k práci. (Vybíhal, 2013)

Způsob, jakým se počítají mzdy, je možné vidět v tabulce Tabulka 1.

Hrubá mzda (HM)	Základní mzda + odměny + příplatky + prémie + osobní ohodnocení
+ Hrazené zdravotní pojištění	9 % z HM, zaokrouhlení na koruny nahoru.
+ Hrazené sociální pojištění	25 % z HM, zaokrouhlení na koruny nahoru.
= Super hrubá mzda (SHM)	SHM se zaokrouhluje na stovky nahoru.
Daň před slevami	SHM * 15 %
Čistá mzda	
- Daň po slevách	Daň před slevami - slevy
- Placené zdravotní pojištění	4,5% z HM
- Placené sociální pojištění	6,5% z HM
Částka k výplatě	
+ Náhrada mzdy	
- Ostatní srážky	

Tabulka 1: Výpočet čisté mzdy (vlastní zpracování)

2.3.2. Zdravotní pojištění

Zdravotní pojištění (ZP) je při výpočtu mezd rozděleno do dvou složek, hrazené a placené ZP. Hrazené ZP je odvedeno za své zaměstnance zaměstnavatel, a to ve výši 9 % z HM. U placené ZP je zaměstnanci strhnuta částka v hodnotě 4,5% z HM. Tyto dvě částky je zaměstnavatel povinen odvést zdravotní pojišťovně zaměstnance. Zaměstnanec se tak nemusí starat o zdravotní pojištění, je mu pouze vyplacena nižší mzda.

Zákonem č. 48/1997 Sb. jsou vymezeny osoby, které musí být povinně pojištěny. Jedná se o osoby, jež mají trvalý pobyt v České republice (ČR), ale i osoby bez trvalého pobytu v ČR zaměstnané u zaměstnavatele se sídlem nebo trvalým pobytem v ČR. Plátcem ZP může být pojištěnec, zaměstnavatel a stát. Pojištěnec je plátcem ZP, pokud je zaměstnancem, osobou samostatně výdělečně činnou (OSVČ) nebo má v ČR trvalý pobyt. Zaměstnavatel je považován za plátce, když je to fyzická (FO) nebo právnická osoba (PO), která je plátcem daně z příjmů ze závislé činnosti a z funkčních požitků (zákon č. 586/1992 Sb., o daních z příjmů) a sídlo (PO) a trvalý pobyt (FO) je v ČR. Stát platí ZP např. za nezaopatřené děti, za ženy na mateřské dovolené. (Vybíhal, 2013)

2.3.3. Pojistné na sociální zabezpečení

Pojistné na sociální zabezpečení (PSZ) je od ZP odlišné pouze v sazbách. Zaměstnavatel přispívá na PSZ částkou 25% z HM a zaměstnanci je odečteno 6,5% z HM. Podle Vybíhala (2013) je hrazené PSZ složeno z příspěvku na státní politiku zaměstnanosti, důchodové a nemocenské pojištění. U placeného PSZ je částka zahrnuta pouze na důchodové pojištění. Poplatníci PSZ jsou opět zaměstnanci, zaměstnavatelé, OSVČ, ale mohou to být také dobrovolníci, kteří si chtějí přispívat na důchodové pojištění.

Hrazené ZP i PSZ představuje „neviditelný příjem zaměstnance“, který podléhá dani z příjmů FO ze závislé činnosti, proto jsou připočteny k HM. Konečná částka i s HM se nazývá super hrubá mzda (SHM).

2.3.4. Daň z příjmů FO ze závislé činnosti a funkčních požitků

Daň z příjmů FO je vypočtena sazbou 15% ze zaokrouhlené SHM (pravidelné i nepravidelné příjmy + 34% z pojistného – ZP a PSZ). Jednotlivé měsíční platby tvoří zálohy, které jsou na konci ročního zdaňovacího období zúčtovány. Zaměstnavatel pravidelně odvádí příslušnému finančnímu úřadu za své zaměstnance zálohy na daň z příjmů. V této situaci je zaměstnavatel plátcem daně a zaměstnanec poplatníkem.

Vypočtená daň může být snížena o slevy na dani (§ 35ba). Měsíční slevy:

- poplatník – 2 070 Kč,
- dítě – 1 117 Kč,
- dítě (držitel průkazu ZTP) – 1 934 Kč,
- invalidita I. a II. stupně – 210 Kč,
- invalidita III. stupně – 420 Kč,
- držitel průkazu ZTP/P – 1 345 Kč,
- student – 335 Kč. (Kučerová [online], 2014)

Je možné dosáhnout i daňového bonusu, kdy finančnímu úřadu vznikne závazek vůči zaměstnanci. Tento bonus lze získat výhradně ze slevy na dítě, podle Kučerové (2014) je to max. do 5 025 Kč. Ostatní slevy snižují daň pouze k nule, kdy nikomu nevzniká žádná povinnost platit, či vracet. Pokud jsou tyto slevy uplatněny u měsíčních záloh, již je nelze znovu použít při ročním zúčtování.

3. Náhled současné situace v podniku

Zde bude přiblížena činnost firmy, její aktuální stav, důvod vytvoření databázové aplikace a požadavky, které jsou na ni kladené.

3.1. O firmě

Firma ČeSi, s. r. o. s místem vykonávání podnikatelské činnosti v Mostech u Jablunkova, pro kterou bude navrhnutá databázová aplikace, se zabývá koupí zboží od různých dodavatelů za účelem jeho následného prodeje koncovým zákazníkům. Jedná se o společnost s ručením omezeným dvou zakladatelů, jež jsou vlastníky dvou maloobchodních jednotek a hotelu s gastronomickými službami.

Informační systém bude projektován na jednu z obchodních jednotek vystupující pod názvem Esíčko. Vybrána byla z důvodu otestování aplikace a bude-li aplikace přínosem pro provoz první jednotky, bude ji tak možné zavést i do druhé jednotky. Obě jednotky vystupují jako samostatné, na sobě nezávislé podniky, tudíž je toto uskutečnitelné. Esíčko bylo vybráno z důvodu, že majitelé v této budově sídlí a budou mít možnost důkladnější kontroly nad systémem.

Jedná se o firmu s deseti zaměstnanci. K prodeji je určena široká škála zboží, od potravin (maso a uzeniny, zelenina, mléčné produkty a mnoho dalšího) až po průmyslové zboží (železářství, elektro, drogerie, papírnictví, kuchyňské vybavení, textil, atd.).

3.2. Současná situace

Jediná aplikace v podniku je zastoupena účetním systémem, který je majiteli používán k zaúčtování všech peněžních transakcí. Majiteli je také vedena evidence zaměstnanců v programu MS Excel, jež slouží k uchování osobních a mzdových informací. MS Excel je rovněž používán k počítání mezd.

Dále proces, jakým dochází k objednávání chybějícího zboží, je také velice neefektivní. Dodavatelů zásobujících tuto prodejnu je několik desítek a každý má rozmanitý sortiment. Proto mají objednávky různý charakter. Jsou prováděny telefonicky, což je časově značně náročné, navíc v případě nesrovnalostí v dodavatelem doručeném zboží je nedohledatelné, zda za chybu může objednavatel, či dodavatel (byly by potřeba záznamy z telefonních rozhovorů). Objednávka je nyní prvně sepsána do papírové podoby a až poté nahlášena příslušnému dodavateli. Další způsob objednávání je vyhledávání zboží v několika seti stránkové sítině, kde je velmi obtížné a zdlouhavé najít konkrétní druh zboží.

Poslední podoba objednání zboží je přímá komunikace s dealerem. Zaměstnanec buď hlásí, jaké zboží chybí a dealer si zapisuje, nebo dealer nabízí svůj sortiment. Tyto všechny postupy jsou zbytečně časově náročné. Podoba původní objednávky je postupně sepisována do sešitu. Pomocí aplikace bude možné od příslušného dodavatele vybrat zboží, uložit ho do objednávky a odeslat.

Zaměstnanci pracují na šesti hodinovou pracovní směnu. Týdenní pracovní doba je nastavena od sedmi ráno do šesti odpoledne. Otevřeno je také v sobotu od sedmi ráno do dvanácti. Zaměstnanci pracují na tři týdenní směny, které jsou pravidelně po třech týdnech střídány. První týden jsou v práci od pondělí do soboty, kde od pondělí do pátku je směna od sedmi do dvou odpoledne a v sobotu od sedmi do dvanácti. Druhý týden je „zkrácený“, tam musí být zaměstnanec celou pracovní dobu, ale pouze v úterý, ve čtvrtek a v sobotu. V třetím „dlouhém“ týdnu jsou odpracovány celé dny v pondělí, ve středu a v pátek. Přes týden jsou placeni standardní sazbou, která byla stanovena zaměstnavateli. Za práci o víkendu dostávají příplatek.

Dovolená je, podle zákona č. 262/2006 Sb., zákoníku práce, v běžném rozsahu čtyř týdnů, jež musí být předem nahlášena a schválena. Dovolená by měla být během roku celá vyčerpána, ale zaměstnavatel to nestanovuje jako nutnou podmínkou. Musí však být využita do posledního dne v březnu následujícího roku

3.3. Požadavky

Byl proveden rozhovor s majiteli prodejny a s vybranými pracovníky, aby byly vymezeny co nejpřesnější nároky na aplikaci.

Jeden z požadavků na systém je, že před vstupem do aplikace bude přihlašovací okno, kde pomocí přihlašovacího jména a hesla bude umožněn přístup do aplikace. Zaměstnanci budou podle práv pro práci s databází rozdělení do dvou skupin. První skupině bude povoleno zpracovávat pouze část systému zabývající se objednávkami a správou dodavatelů. Druhá skupina bude zahrnovat práva první skupiny, ale navíc bude mít na starosti mzdový systém a evidenci zaměstnanců. Do této skupiny patří vedoucí prodejny, kteří nyní v papírové podobě zaznamenávají odpracovanou dobu jednotlivých zaměstnanců, pracovní neschopnost a využitou dovolenou.

V objednávkovém systému bude zaznamenáváno veškeré dostupné zboží, které bude možné mazat, upravovat a přidávat nové. K rychlému a jednoduchému vyhledávání zboží je vyžadováno fulltextové vyhledávání, protože by bylo velice neefektivní prohledávat

jednotlivé záznamy zboží. Formuláře by měly být realizovány tak, aby se s konkrétním dodavatelem zobrazilo jim nabízené zboží a podle této nabídky je pak vytvořena objednávka. Objednávku nebude nutné ihned odeslat, ale bude možno ji postupně ukládat a modifikovat a při konečné verzi ji odeslat, či vytisknout.

3.4. Ostatní

Data mohou být z reality získávána různými způsoby a z různých interních, či externích zdrojů. Pro nalezení a uložení všech potřebných dat do databáze budou využity veškeré možné zdroje, které v podniku již existují v papírové formě. Dále budou zpracovány poskytnuté rozhovory s majiteli obchodní jednotky a všemi jejími zaměstnanci a kontaktování budou i dodavatelé.

Samozřejmě bude brán zřetel na nové uživatele aplikace, kteří s ní přijdou poprvé do styku. V této situaci mohou vzniknout jakékoli chyby z neznalosti, ale i chyby vzniklé nedopatřením. Snahou bude veškeré možné chyby ošetřit a vyhnout se tak problémům např. pole vyplněno nesprávnou hodnotou nebo nevyplnění pole, kde je to nutné.

Majitelé byli srozuměni s tím, že podobnou aplikaci by na trhu jistě našli, ale museli by si připlatit za implementaci i za případnou správu a aktualizace. Proto bude aplikace vyvinuta v rámci bakalářské práce. Aplikace bude odpovídat veškerým požadavkům. Tímto bude šitá na míru vybranému podniku a všechny důležité prvky aplikace budou přehledné a jednoduše dostupné. Pro tento malý podnik bude tato navrhnutá databáze plně dostačující.

4. Návrh databázové aplikace a implementace

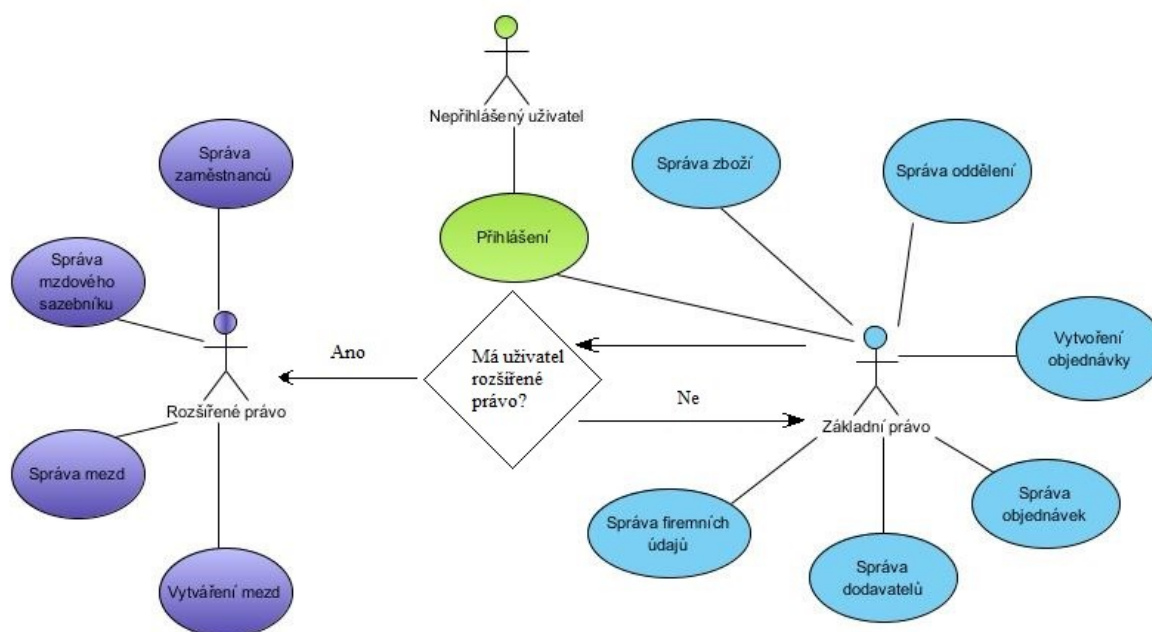
Základním krokem při návrhu informačního systému je vyčlenit hlavní objekty reality, které jsou pro práci s aplikací nutné a vyžadující. Objekty jsou pouze informačního charakteru, aby bylo zřejmé, s jakými entitami a jejich atributy bude databáze manipulovat.

V dalším kroku jsou tyto objekty převedeny do skutečných tabulek, již v prostředí programu MS Access 2010 s konkrétními sloupci a jejich vlastnostmi. Musí být jednoznačně stanoven primární klíč každé vytvořené tabulky, příp. i cizí klíč. S polem je asociován jeho datový typ, který byl zmíněn v teoretické části práce (*Automatické číslo, Text, Číslo, Pole se seznamem, atd.*). Je nastavena jeho velikost a/nebo formát, eventuálně vstupní maska. Je určeno, zda musí být hodnota pole zadána a také zda bude indexována. Při zhotovení všech tabulek jsou naznačeny veškeré vztahy, jež mezi tabulkami vznikají.

Po vytvoření tabulek jsou zpracována data, která se už při implementaci aplikace do podniku musí v databázi prakticky nacházet. Tato data byla získána od zaměstnanců a majitelů firmy ČeSi, s. r. o. Pro odzkoušení funkčnosti aplikace je do systému zavedena pouze malá část dat, která k realizaci plně postačuje.

Další částí je navrhnout funkční aplikace pomocí dotazů, formulářů a sestav s využitím programovacího jazyka Visual Basic for Application a maker. Na přání žadatele je databáze chráněna proti neoprávněnému přístupu. Po přihlášení mají všichni uživatelé přístup k základním funkcím aplikace a uživatelé, kteří jsou vedením firmy pověřeni, i k funkcím rozšířeným, jak je znázorněno na obrázku 7 pomocí Use case diagramu. Obrázkem je také naznačeno barevné schéma, jaké je u jednotlivých formulářů použito. K jednoduchému ovládání aplikace je vytvořen ve formuláři pás karet jako navigační okno, kterým je možnost se dostat ke každému formuláři v databázi (objednávkový, mzdový a evidenční systém). Prostřednictvím sestav je vytvořen návrh objednávky a výplatního pásku, aby tyto dva dokumenty bylo možné vytisknout či odeslat.

V posledním kroku budou nahrána do databáze všechna data. Aplikace bude uložena na přenositelné médium (CD, Flash, ...) a bude provedena její implementace na podnikovou stanici. Bude odzkoušeno její používání v praxi, doladěny nesrovnalosti a školení uživatelů.



Obrázek 7.: Use case diagram (UML) – diagram použití

4.1. Popis objektů reality

Po prozkoumání chování v podniku se došlo k závěru, že je nutné zahrnout deset objektů pro celou databázi.

Zboží

Tento objekt je poskytovatelem informací o zboží v prodejně – název, dodavatel, oddělení.

Oddělení

Je nositelem informací o názvu oddělení a sazbě DPH.

Dodavatelé

Poskytuje základní údaje o dodavateli. Název dodavatele, IČO, kontaktní údaje – ulice, město, PSČ, telefon, mobil, fax a e-mail.

Objednávka

Zaznamenává datum vyhotovení objednávky, dodavatele a odpovědnou osobu.

Objednané zboží

Tento objekt je poskytovatelem množství objednaného zboží, jednotkách, ve kterých je zboží vedeno a orientační ceně.

Firma

V tomto objektu jsou obsaženy údaje o firmě, pro kterou je databáze vytvářena. Název, kontaktní a bankovní údaje (název banky, předčíslí a číslo účtu, kód banky).

Zaměstnanci

Objekt je uchovatelem základních informací o zaměstnancích. Jméno, příjmení, přihlašovací jméno a heslo, rodné číslo, kontaktní údaje (ulice, město, PSČ, telefon, mobil), profesi vykonávající v podniku s hodinovou sazbou. Zda je poplatníkem, studentem nebo ZTP, jestli není invalidou 1., 2., či 3. stupně a na kolik dětí (i ZTP) může uplatnit slevu na dani (může pouze jeden z rodičů).

Mzdové údaje

Tento objekt je nositelem údajů o období počítání mzdy, data výpočtu a zaměstnanci, kterému je mzda počítána.

Vypočtené mzdy

Tento objekt je uložštěm odpracované doby, přesčasu, odměny, osobního ohodnocení, hrubé mzdy, hrazeného i placeného zdravotního pojištění a sociálního zabezpečení, dani před slevami, celkové slevy, dani po slevách, čisté mzdy, ostatních srážek a příplatků, mzdy k výplatě, nároku na dovolenou a využití dovolené.

Sazebník mezd

Úkolem objektu sazebník mezd je spravovat sazby důležité k výpočtu mezd (poplatník, dítě, dítě s ZTP, invalidita 1., 2. a 3. stupně, ZTP student, sazby hrazeného i placeného zdravotního a sociálního pojištění, sazba daně z příjmů).

4.2. Tabulky a datový slovník

V další části návrhu aplikace je důležitá praktická realizace tabulek. Všechny objekty jsou převedeny všechny objekty do konkrétních tabulek, kde musí být jednoznačně stanoven primární klíč a integrita. Ke zhotoveným tabulkám jsou vymezeny vztahy, v jakých k sobě jednotlivé tabulky vystupují.

Podrobně budou rozebrány jednotlivé tabulky pomocí datového slovníku.

4.2.1. Tabulka tblOddeleni

Tabulka tblOddeleni bude uchovávat údaje o všech odděleních, na které je podnik členěn. Oddělení je vedeno tak, aby ulehčilo práci při vyhledávání zboží a je také poskytovatelem hodnoty DPH. Každé oddělení má svou sazbu DPH, tak jak je dáno daňovým zákonem. Primární klíč je u pole *IDOddeleni* jako automatické číslo. Jelikož firmě může odpovídat několik oddělení, je zde obsažen cizí klíč *KodFirmy*, jež odpovídá primárnímu klíči z tabulky tblFirma. V *Oddeleni* (30) je veden název oddělení a v *DPH* je zaznamenávána sazba DPH jako procento s jednoduchou přesností, jak je ukázáno v datovém slovníku tabulky tblOddeleni v tabulce 1.

Klíč	Název pole	Datový typ	Velikost Formát	Vstupní maska	Nutno zadat	Indexovat
PK	IDOddeleni	Aut. Cislo	Dlouhé celé číslo		-	Bez duplicity
FK	KodFirmy	Text		LLLL0000		
	Oddeleni	Text	30		Ano	Ne
	DPH	Číslo	Jednoduchá přesnost/Procenta		Ano	Ne

Tabulka 1.: Design tabulky tblOddeleni

4.2.2. Tabulka tblDodavatel

Tabulka tblDodavatel bude uchovávat údaje o veškerých dodavatelích, se kterými je podnik v kontaktu. Primární klíč je nastaven na sloupci *KodDod*, který je v textovém formátu o velikosti osm znaků se vstupní maskou „LLLL0000“. Tento musí být zadán a údaje budou indexovány. Ostatní sloupce budou rovněž datového typu text. Jedná se o název firmy *Firma* (30), *ICO* (15), *Ulice* (30), *Mesto* (30), *PSC* (5), vzhled PSČ bude automaticky generován pomocí vstupní masky „000\ 00“ (např. 739 98). *Telefon*, *Mobil* i *Fax* (9) jsou také tvořeny vstupní maskou „000\ 000\ 000“. Podrobně je to znázorněno v tabulce 2.

Klíč	Název pole	Datový typ	Velikost	Vstupní maska	Nutno zadat	Indexovat
PK	KodDod	Text	8	LLLL0000	Ano	Bez duplicity
	Firma	Text	30		Ano	Ne
	ICO	Text	15		Ne	Ne
	Ulice, Mesto	Text	30		Ano	Ne
	PSC	Text	5	000\ 00	Ano	Ne
	Telefon	Text	9	000\ 000\ 000	Ne	Ne
	Mobil	Text	9	000\ 000\ 000	Ano	Ne
	Fax	Text	9	000\ 000\ 000	Ne	Ne
	Email	Text	25		Ne	Ne

Tabulka 2.: Design tabulky tblDodavatel

4.2.3. Tabulka tblZbozi

Tabulka tblZbozi je úložištěm pro všechno zboží, kterým podnik prosperuje a nabízí jej zákazníkům ke koupi. Jako primární klíč je zde nastaven *KodZbozi* (10), je datového typu text a vzhled je generován vstupní maskou „9999999999“, hodnota pole musí být vyplněna a nesmí zůstat prázdná. Další pole v tabulce je *Nazev* (50) a bude zaznamenávat název daného zboží. V této tabulce se nacházejí cizí klíče, které jsou prezentovány v tabulce 3. Cizí klíč *Dodava* (8) představuje pole, které identifikuje totožnost dodavatele, jež dodává zboží. Další cizí klíč *IDOddeleni* určuje k jakému oddělení je zboží řazeno. Oba cizí klíče musí být zadány, hodnota pole nesmí být Null a indexovány jsou s duplicitou.

Klíč	Název pole	Datový typ	Velikost/Formát	Vstupní maska	Nutno zadat	Indexovat
PK	KodZbozi	Text	10	9999999999	Ano	Bez duplicity
	Nazev	Text	50		Ano	Ne
FK	Dodava	Text	8	LLLL0000	Ano	S duplicitou
FK	IDOddeleni	Číslo	Dlouhé celé číslo		Ano	S duplicitou

Tabulka 3.: Design tabulky tblZbozi

4.2.4. Tabulka tblObjednavka

Tabulka tblObjednavka bude sloužit k ukládání vyhotovených objednávek jak rozpracovaných, tak i odeslaných. Primární klíč je zde *IDObj* textového formátu o velikosti

osm znaků a vstupní maskou „99999999“ opět proto, že nemusí být vyplněny číslice na všech pozicích. Toto pole je při zadávání dat do databáze povinné. *DatVyhotoveni* datového typu Datum a čas je rovněž povinný údaj a při zhotovení objednávky se automaticky vytvoří s použitím funkce Date(). V tabulce tblObjednavka se vyskytují tyto cizí klíče: *Login* (8) a *Dodavatel* (8), jak je naznačeno v tabulce 4. *Login* je primárním klíčem tabulky tblZamestnanec a v této tabulce určuje osobu, která je zodpovědná za vytvoření objednávky. Je to sice povinný údaj, ale bude se zobrazovat automaticky podle toho, který uživatel je momentálně přihlášený a se systémem pracuje. *Dodavatel* jako primární klíč tabulky tblDodavatel jednoznačně determinuje, u kterého dodavatele je objednávka objednána. Oba tyto cizí klíče jsou indexovány s duplicitou a mohou se tak v tabulce objevovat několikanásobně.

Klíč	Název pole	Datový typ	Velikost Formát	Vstupní maska	Nutno zadat	Indexovat
PK	IDObj	Text	8	99999999	Ano	Bez duplicity
	DatVyhotoveni	Datum/čas	Datum		Ano	Ne
FK	Login	Text	8	LLLL0000	Ano	S duplicitou
	Platba, Stav	Text			Ano	Ne
FK	Dodavatel	Text	8	LLLL0000	Ano	S duplicitou

Tabulka 4.: Tabulka tblObjednavka

4.2.5. Tabulka tblObjZbozi

V tabulce tblObjZbozi bude zaznamenáváno veškeré zboží, které bylo objednáno. Primární klíč je automatické číslo s názvem *IDObjZb*, jež se samočinně generuje při každém nově vloženém záznamu. Cizí klíče v této tabulce jsou *IDObj* (8) z tabulky tblObjednavky a *KodZbozi* (10) z tabulky tblZbozi. *IDObj* zde identifikuje, k jaké objednávce objednané zboží náleží, proto musí být nutně zadáno. Pomocí *KodZbozi* je přesně dáno o jaké zboží se jedná. Indexace je u obou cizích klíčů opět s duplicitou. V *Mnozstvi* je datovým typem číslo určen objem objednaného zboží a polem *Jednotky* je definováno, v jakých jednotkách je zboží vedeno. *OrientCenaDPH* má pouze informativní charakter, aby zaměstnanci přibližně věděli, kolik za objednávku celkem zaplatí. Přesnou cenu nebylo možné zavést, protože každý dodavatel má cenu různou a tato je průběžně měněna. Proto nákupní cena bude uvedena až na faktuře, kterou vystaví dodavatel. Cena se nachází v tabulce tblObjZbozi, místo v tabulce tblZbozi, protože cena souvisí s konkrétním objednaným zbožím a tím, že cena bude v tabulce

tblObjZbozi zůstane po celou dobu neměnná, kdežto v tabulce tblZbozi by při změně ceny, došlo k přepočítání všech předešlých objednávek. V tabulce 5 je vidět přesný design tabulky tblObjZbozi.

Klíč	Název pole	Datový typ	Velikost/Formát	Vstupní maska	Nutno zadat	Indexovat
PK	IDObjZb	Aut. Cislo	Dlouhé celé číslo		-	Bez duplicity
FK	IDObj	Text	8	99999999	Ano	S duplicitou
FK	KodZbozi	Text	10	9999999999	Ano	S duplicitou
	Mnozstvi	Číslo	Dlouhé celé číslo		Ano	Ne
	Jednotky	Text			Ano	Ne
	OrientCenaDPH	Měna	Měna		Ne	Ne

Tabulka 5.: Design tabulky tblObjZbozi

4.2.6. Tabulka tblZamestnanec

V tabulce 6 je možné vidět design tabulky tblZamestnanec, který bude uchovávat záznamy o zaměstnancích pracujících v prodejně. Primární klíč je nastaven u pole *Login* (8) v textovém formátu a vstupní maskou „LLLL0000“, kde první čtyři písmenka jsou začáteční písmena z příjmení a čtyři číslice jsou poslední čtyřčíslí z rodného čísla. Není to však nutnou podmínkou, každý zaměstnanec si může zvolit své přihlašovací jméno. Nesmí se však shodovat s žádným jiným, což je ošetřeno, jak indexováním bez duplicity, tak také v samotných formulářích při přidávání nového zaměstnance. Tento primární klíč bude rovněž sloužit jako údaj při přihlašování do systému. Ke každému zaměstnanci musí být, kromě *Telefon* (9), povinně vyplněno *Jméno* (20), *Příjmení* (30) a *RodCislo* (10), kde rodné číslo má nastavenou vstupní masku na „000000"/"0000“. Dále musí mít zaměstnanci své heslo, které spolu s loginem umožní práci s aplikací. *Heslo* je textového formátu s minimální délkou šest a maximální patnáct znaků. Zaměstnanci je rovněž přidělena profese, kterou provádí a má nastaveno přístupové právo základní, či rozšířené. Zbývající pole se týkají mzdových a kontaktních údajů. Mzdové údaje určují, zda má zaměstnanec možnost uplatnit slevu na dani z příjmů a příp. na kolik dětí může slevu uplatnit. Těmto polím odpovídá vždy jedna sazba z tabulky tblSazbyMezd, kde vzniklo propojení polem *IDSazby*, jemuž v tabulce tblZamestnanec odpovídá cizí klíč. Kontaktní údaje zaznamenávají adresu bydliště a telefonní kontakt.

Klíč	Název pole	Datový typ	Velikost/ Formát	Vstupní maska	Nutno zadat	Indexovat
PK	Login	Text	8	LLLL0000	Ano	Bez duplicity
	Jmeno	Text	20		Ano	Ne
	Prijmeni	Text	30		Ano	Ne
	RodCislo	Text	10	000000"/"0000	Ano	Ne
	Heslo	Text	15	Heslo	Ano	Ne
	Profese	Text			Ano	Ne
	HodSazba	Měna	Měna		Ano	Ne
	Pravo	Ano/Ne			-	Ne
FK	IDSazby	Číslo	Dl. celé číslo		Ano	S duplicitou
	Poplatnik	Ano/Ne			-	Ne
	PocetDeti, PocetDetiZTP	Pole se seznamem			Ano	Ne
	Invalida1_2, Invalida3, ZTP, Student	Ano/Ne			-	Ne
	Ulice, Mesto	Text	30		Ano	Ne
	PSC	Text	5	000\ 00	Ano	Ne
	Telefon	Text	9	000\ 000\ 000	Ne	Ne
	Mobil	Text	9	000\ 000\ 000	Ano	Ne

Tabulka 6.: Design tabulky tblZamestnanec

4.2.7. Tabulka tblMzdoveUdaje

Tabulka tblMzdoveUdaje bude zahrnovat záznamy z výpočtu mezd jednotlivých zaměstnanců v daném období. Primární klíč v této tabulce je zastoupen *IDMzdy*, jak lze vidět v tabulce 7. Toto pole je datového typu text o velikosti šestnáct znaků a vstupní maskou „LLLL0000": "90"/"0000“. Tato vstupní maska je složena z přihlašovacího jména, aby byli zaměstnanci od sebe odlišeni a z období, v němž je mzda počítána, tudíž od sebe odlišuje každou vypočtenou mzdu a tím je zachována její jedinečnost. Údaj musí být vyplněn a hodnota pole nesmí zůstat prázdná. Dalšími poli je výše zmíněné *Obdobi* (6) se vstupní maskou „90\0000“, kde první dvě číslice představují měsíc a zbylé čtyři rok. Pole *DatumVypoctu* je datového typu Datum a čas. Tyto dva údaje musí být zadány.

Klíč	Název pole	Datový typ	Velikost Formát	Vstupní maska	Nutno zadat	Indexovat
PK	IDMzdy	Text	16	LLLL0000": "90"/"0000	Ano	Bez duplicity
	Obdobi	Text	6	90V0000	Ano	Ne
	DatumVypoctu	Datum/čas	Datum		Ano	-

Tabulka 7.: Tabulka tblMzdoveUdaje

4.2.8. Tabulka tblVypocteneMzdy

Tabulka 8 znázorňuje přibližný vzhled tabulky tblVypocteneMzdy. Tato tabulka poskytuje prostor pro uložení všech vypočtených mezd pro každého zaměstnance v každém období výpočtu. Primární klíč je automatické číslo *IDVypoctu*, jež je generováno systémem. Tabulka také obsahuje cizí klíče. *Login* identifikuje zaměstnance. Přes pole *IDMzdy* z tabulky tblMzdoveUdaje, z které je možné zjistit období a datum výpočtu mzdy. Kromě tří výše zmíněných polí musí být povinně zadána ještě *OdpracDoba*. Nepovinně může zaměstnanec zadat *Prescas*, *Odmena*, *OsOhodnoceni*, *OstSrazky*, *OstPripl* a *VyuzDov*. Zbývající jsou doplněny samy naprogramovaným výpočtem.

Klíč	Název pole	Datový typ	Velikost Formát	Vstupní maska	Nutno zadat	Indexovat
PK	IDVypoctu	Aut. Cislo	Dlh. celé číslo		-	Bez duplicity
FK	Login	Text	8	LLLL0000	Ano	S duplicitou
FK	IDMzdy	Text	16	LLLL0000": "90"/"0000	Ano	S duplicitou
	OdpracDoba	Číslo	Dlh. celé číslo		Ano	Ne
	Prescas	Číslo	Dlh. celé číslo		Ne	Ne
	Odmena, OsOhodnoceni, Hrubá, ZdravotniHr, SocialniHr, ZdravotniPl, SocialniPl, DanPred, SlevaCelkem, DanPo, Cista, OstSrazky, OstPripl					
	Kvyplate	Měna	Měna		Ne	Ne
	NarokDov, VyuzDov	Číslo	Dlh. celé číslo		Ne	Ne

Tabulka 8.: Design tabulky tblVypocteneMzdy

4.3. Kardinalita vztahu mezi tabulkami

Na obrázku 8 jsou znázorněny veškeré vztahy, které se v databázi mezi jednotlivými tabulkami nacházejí.

tblMzdoveudaje : tblVypocteneMzdy – 1 : 1

Vztah u těchto tabulek znamená, že každému mzdovému údaji odpovídá jen jedna vypočtená mzda.

Ostatní tabulky se vyskytují už jen ve vztahu 1 : N, kdy např. jeden zaměstnanec bude odpovídat za několik vytvořených objednávek, ale k jedné objednávce je přiřazen pouze jeden zodpovídající zaměstnanec.

V prvních dvou vztazích je možné vidět situaci, kdy prakticky nelze vytvořit vztah M : N, a tabulka tblObjZbozi slouží jako spojovací tabulka mezi tabulkami tblObjednavka a tblZbozi. Protože v různých objednávkách se zboží může vyskytovat několikanásobně a na druhou stranu zboží může být součástí několika vyhotovených objednávek.

tblObjednavka : tblObjZbozi – 1 : N

tblZbozi : tblObjZbozi – 1 : N

tblDodavatel : tblObjednavka – 1 : N

tblDodavatel : tblZbozi – 1 : N

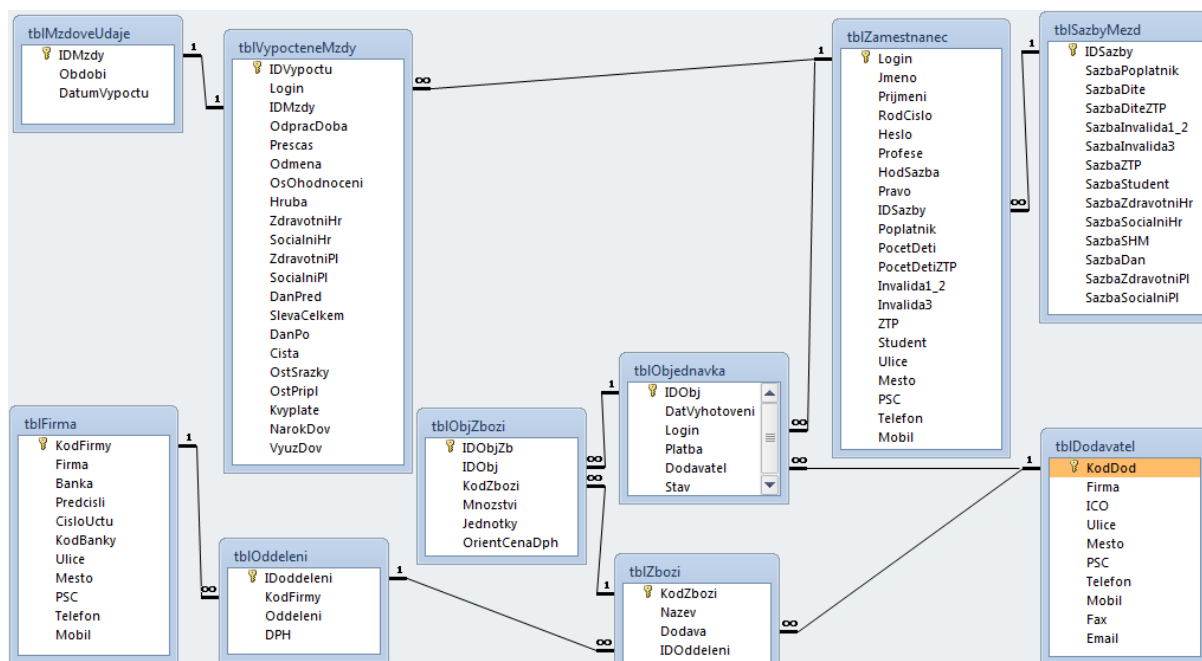
tblOddeleni : tblZbozi – 1 : N

tblFirma : tblOddělení – 1 : N

tblZamestnanec : tblVypocteneMzdy – 1 : N

tblZamestnanec : tblSazbyMezd – 1 : N

tblZamestnanec : tblObjednavka – 1 : N



Obrázek 8.: Ukázka vztahů mezi jednotlivými tabulkami

4.4. Aplikace

Po prozkoumání všech objektů vyskytujících se v realitě, až po skutečné navržnutí všech tabulek, je pomocí formulářů sestrojena aplikace tak, aby uživatel nemusel pracovat s nesympatickým prostředím pouhých tabulek.

4.4.1. Zabezpečení

Jelikož se v databázi objevují choulostivá data, ke kterým mohou přístup pouze vybraní zaměstnanci, jež mají právo s těmito daty pracovat, muselo být toto zajištěno. Všem zaměstnancům musí být přiděleno přihlašovací jméno a heslo umožňující práci s databází. Tímto mohlo být také dosaženo toho, že jsou zaměstnancům nastavena přístupová práva.

4.4.2. Formulář pro přihlášení

Formulářem na obrázku 9, který se zobrazí hned po rozběhnutí aplikace, je možné přistoupit k samotné aplikaci. Přihlášení je ošetřeno jak proti nezadanému jménu a heslu, tak i proti nesprávnému loginu a heslu uložených v tabulce tblZamestnanec.

Tlačítkem *Ukončit* dojde k zavření celé aplikace.

Obrázek 9.: Formulář frmPrihlaseni – Přihlašovací okno

4.4.3. Formulář základního okna

Po úspěšném přihlášení se zobrazí základní okno, viz obrázek 10, které slouží jako místní nabídka – navigační okno. Focus je nastaven na úvodní stránku zobrazující aktuálně přihlášeného uživatele. Na této stránce se rovněž může uživatel odhlásit, což zobrazí prázdné přihlašovací okno, nebo druhým tlačítkem zavře aplikaci.

Pás karet se liší podle toho, zda má uživatel nastaveno právo základní, či rozšířené. Jestli má uživatel pouze právo základní, může spravovat pouze zboží, oddělení, nové i již vytvořené objednávky a údaje o firmě. Pokud má právo rozšířené, zobrazí se mu celý pas karet, který slouží k evidenci zaměstnanců, k vedení nových i vypočtených mezd a k úpravě sazebníku.

Obrázek 10.: Formulář frmZakladniOkno – Základní a rozšířené právo

V další části bude charakterizována každá stránka z pásu karet.

Základní právo:

- správa zboží,
- správa oddělení,
- vytváření objednávky,
- správa objednávek,
- správa dodavatelů,
- správa firemních údajů.

Rozšířené právo:

- vše ze *základního práva*,
- správa mzdového sazebníku,
- správa mezd,
- vytváření mezd.
- správa zaměstnanců.

Všechny formuláře, které se týkají pouze vybraného okruhu lidí, jsou odlišeny fialovou barvou. Formuláře, jež jsou dostupné každému uživateli, mají modrou barvu.

4.4.4. Formuláře pro manipulaci se zbožím

Na kartě *Zboží* je vedena evidence veškerého zboží v podniku. Vzhled tohoto formuláře je znázorněn na obrázku 11. Pro ulehčení při vyhledávání konkrétního zboží byla do formuláře připojena funkce, která vyfiltruje takové údaje, jež odpovídají podmínce. Vyhledávat se může podle oddělení, druhu zboží, nebo dodavatelů. V případě, že uživatel bude chtít znovu vše zobrazit, může k tomu využít tlačítko *Zobrazit vše*.

```
Private Sub btnVyhledatZbozi_Click()  
    If IsNull(txtVyhledatZbozi) Then  
        MsgBox "Pole vyhledat podle zboží je prázdné", vbExclamation  
        txtVyhledatZbozi.SetFocus  
        Exit Sub  
    End If  
    Filter = "[Nazev] Like '*' & txtVyhledatZbozi & '*'"  
    FilterOn = True  
    If Me.RecordsetClone.RecordCount = 0 Then  
        MsgBox "Žádný záznam nevyhovuje.", vbExclamation, "Zboží"  
        FilterOn = False  
        txtVyhledatZbozi.SetFocus  
    End If  
End Sub
```

Úvodní

Zboží

Nová objednávka

Přehled objednávek

Oddělení

Firma

Správa zboží

Kód zboží	Název	Dodavatel	Oddělení	DPH
1	Rajče	HRUS0000	Zelenina	15%
2	Žárovka	EMOS1114	Elektro	21%
3	Jahody	HRUS0000	Ovoce	15%
4	Rohlík	BAJO1112	Pečivo	15%
6	Jogurt	HRUS0000	Mléčné výrobky	15%
7	Vypínač	EMOS1114	Elektro	21%
8	Baterie AAA	BATY1111	Elektro	21%
9	BaterieAA	BATY1111	Elektro	21%
10	Baterie 9V	BATY1111	Elektro	21%

Obrázek 11.: Karta zboží

Na kartě zboží uživatel může přidat nové zboží a upravovat stávající. Pokud uživatel zadá kód zboží, který v tabulce již existuje, zobrazí se chybové hlášení. Ošetřeny jsou také nevyplněné a nevybrané pole. V obou tabulkách, zobrazených na obrázku 12, musí uživatel přes další dva formuláře vybrat dodavatele a oddělení. Uživatel v seznamu nalezne správné údaje a ten odešle kliknutím na příkazové tlačítko *Vybrat*. Následně se do pole *Dodavatel a Oddělení* vypíše identifikační kód charakterizující výběr.

Přidat zboží

Úprava zboží 1

Přidejte nové zboží

Kód zboží

Název

Dodavatel

Oddělení

Úprava zboží

Kód zboží

Název

Dodavatel

ID Oddělení

Obrázek 12.: Manipulace se zbožím – přidávání, úprava

Pokud uživatel bude chtít odstranit záznam, bude záležet, zda je zboží součástí rozpracované objednávky. V případě rozpracované objednávky nebude možné údaj odstranit. Pokud se bude jednat o odeslanou objednávku, údaj může být odstraněn.

4.4.5. Formuláře pro novou objednávku

K vytvoření nové objednávky je určena karta *Nová objednávka*, na které je umístěn formulář pro výběr dodavatele (obrázek 13), jemuž je objednávka určena. Tato karta je také vytyčena k evidenci dodavatelů, čili k jejich přidávání, úpravě a odstranění.

Název Ulice	Kód	Mobil Město	Telefon	Fax	E-mail PSČ	IČO
Pekárna - Bajusz ČSA 26	BAJO1112	185 165 126 Jablunkov	818 515 185	612 965 262	bajusz@seznam.cz 739 97	12345678
Baterie Hornopolní	BATY1111	608 547 856 Třinec			baterie1111@gmail.cz 975 25	89658745
EMOS Lipnická 2844	EMOS1114	736 985 365 Přerov	581 261 160	581 261 135	emos@emos.cz 750 02	19014104
Hruška Táboristů 1	HRUS0000	789 652 365 Olomouc	585 119 135	585 119 135	mohruska@volny.cz 779 00	11111111
Žvak Černá cesta 957	ZVAK1113		596 425 588 Horní Suchá	596 425 589	zvak@centrum.cz 735 35	45195552

Přidat dodavatele Upravit dodavatele Odstranit dodavatele

Vyberte dodavatele, pro kterého si přejete vytvořit objednávku

Obrázek 13.: Formulář pro evidenci, modifikaci, smazání a výběr dodavatele

Pokud je uživatelem přidáván nový dodavatel, nesmí vzniknout redundantní kód dodavatele. Tato možnost je ošetřena zobrazením informačního okna a uživatel je povinen tento údaj změnit. Položky, které je nutné vyplnit, jsou označeny hvězdičkou. V případě, že uživatel položku s hvězdičkou nevyplní, je zobrazena hláška.

Při úpravě dodavatele je otevřen formulář pro úpravu, jeho titulek identifikuje, který dodavatel je momentálně upravován, a do polí jsou načteny výchozí hodnoty. Tyto hodnoty může uživatel jakýmkoli způsobem upravovat, musí však opět brát ohled na jedinečný kód dodavatele a na vyplnění všech povinných polí. Po uložení je proveden SQL příkaz a tabulka tblDodavatele je aktualizována. Formuláře pro správu dodavatelů, *Přidat a upravit dodavatele*, jsou zobrazeny na obrázku 14.

Obrázek 14.: Formuláře pro přidání a úpravu dodavatele

Klikem na tlačítko *Vybrat dodavatele*, pro kterého si přejete vytvořit objednávku, se uživateli zobrazí prázdný formulář s nabídkou zboží odpovídající zadanému dodavateli, přesně jako na obrázku 15. Odfiltrování vybraného dodavatele s jeho nabídkou zboží je provedeno následující procedurou, která je zahájena otevřením formuláře *Vyhotovení objednávky*.

```
Private Sub Form_Open(Cancel As Integer)
    Filter = "[Dodava] Like '*' & Form_frmDodavatele.KodDod & '*'"
    FilterOn = True
    If Me.RecordsetClone.RecordCount = 0 Then
        If MsgBox("Dodavatel " & Form_frmDodavatele.KodDod & _
            " neposkytuje žádné zboží!" & vbCrLf & _
            "Přejete si vybrat jiného dodavatele (ANO)." & vbCrLf & _
            "Nebo chcete dodavateli přidat nové zboží (NE) & _
            "?", vbQuestion + vbYesNo) = vbYes Then
            DoCmd.Close acForm, "frmObjednavka"
        End If
    End If
    FilterOn = False
End If
End Sub
```

Před tím, než uživatel začne z nabídky vybírat zboží, musí být objednávka vždy vytvořena. Nastane-li tato situace, je otevřeno okno pro vytvoření objednávky. Při otevření tohoto okna se automaticky doplní datum vyhotovení podle aktuálního data a přihlašovací jméno aktuálního uživatele.

Každé objednané zboží je postupně načítáno do vedlejšího formuláře, kde je možné znovu modifikovat množství a orientační cenu, příp. lze objednané zboží ze seznamu

odstranit. Pro stornování celé objednávky i s již objednaným zbožím je naprogramováno tlačítko *Storno objednávky*.

Tlačítko *Správa zboží* je určeno k úpravě, nebo přidávání zboží v průběhu objednávky. Objednávací formulář je zneviditelněn a je zobrazena karta zboží pro modifikaci. Pro navrácení se zpět k objednávce je pouze v této situaci na kartě zboží zobrazeno tlačítko zpět na objednávku, kterým je navrácen fokus objednávce.

Z formuláře *Nová objednávka* si uživatel může znovu vybrat nového dodavatele, a to tlačítkem *Vybrat nového dodavatele*. Tím dojde k zavření aktivního formuláře a je nastaven fokus na kartu *Nová objednávka*. Na podobném principu funguje také tlačítko *Přehled všech objednávek*, je zavřen aktivní formulář a nastaven fokus na kartu *Přehled objednávek*.

Kód zboží	Množství	Jednotky	Orient. cena	Kód zboží	Název	Dodavatel	Oddělení	DPH
8				8	Baterie AAA	BATY1111	Elektro	21%
9				9	BaterieAA	BATY1111	Elektro	21%
10				10	Baterie 9V	BATY1111	Elektro	21%

Obrázek 15.: Formulář nová objednávka

4.4.6. Formuláře pro přehled a úpravu objednávek

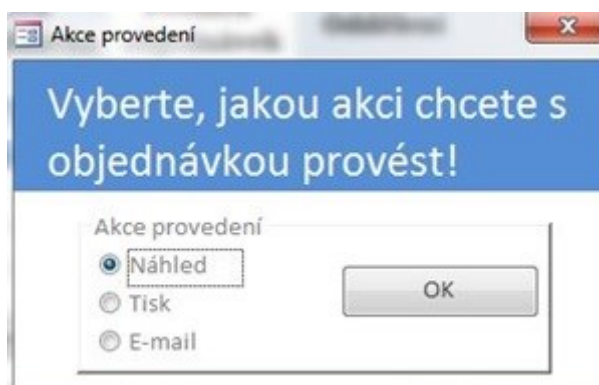
Na kartě přehled objednávek je možnost procházet všechny rozpracované i odeslané objednávky. Z této karty je také možné upravovat rozpracované objednávky. Odeslané objednávky již modifikovat nelze a je u nich hlášena výjimka.

Formulář pro úpravu objednávky je vzhledově totožný s formulářem pro novou objednávku. Při otevření tohoto formuláře jsou do polí načteny hodnoty vybrané objednávky k přepracování. Jsou zde stejné možnosti správy – úprava objednaného zboží, mazání objednaného zboží, přidávání nově objednaného zboží, úprava objednávacích údajů i smazání

celé objednávky. Při úpravě objednávky je přepsáno pole *Zodpovědný* přihlašovacím jménem zaměstnance, který jako poslední provedl aktualizaci objednávky.

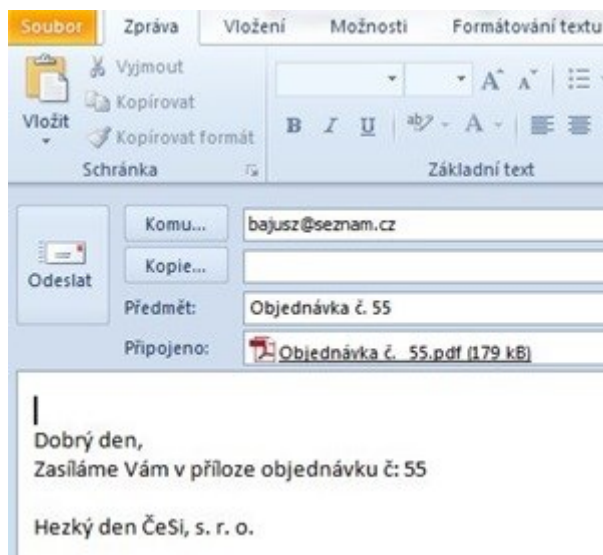
Stisknutím tlačítka *Akce s objednávkou* ve formuláři přehled objednávek se otevře nový formulář, ve kterém je na výběr několik možností – *Náhled*, *Tisk*, *E-mail* – prezentováno na obrázku 16.

Pomocí volby *Náhled* se uživatel může podívat na formu objednávky před vytištěním, nebo odesláním. Sestava má klasickou podobu objednávky, ukázka se nachází v příloze č. 2. Volbou *Tisk* je zobrazen náhled před tiskem a možnost nastavení tisku.



Obrázek 16.: Akce provedení s objednávkou

Posledním přepínačem *E-mail* je zajištěno odeslání objednávky e-mailem příslušnému dodavateli. K tomu je využita aplikace z balíčku MS Office pro práci s e-mailem – MS Outlook. Firma již dříve tuto aplikaci využívala, proto nemusí pořizovat licenci k tomuto softwaru. Na pracovní stanici bude v tomto SW vytvořen e-mailový účet, který zajistí posílání a správu objednávek. Při výběru volby *E-mail* a následném stisknutí tlačítka *OK* je v prvním kroku zjištěno, jestli je na stanici spuštěn MS Outlook. V případě že Outlook spuštěn není, je naprogramováno, aby se stiskem tlačítka *OK* Outlook automaticky spustil. Jakmile je Outlook spuštěn, bude zobrazena nová zpráva s automaticky vyplněným předmětem zprávy *Objednávka č. (IDObj)*, textem zprávy a souborem *Objednávka č. (IDObj).pdf*, ve kterém je uložena samotná objednávka. Ukázka nové zprávy je na obrázku 17. V programu Outlook se do e-mailové adresy příjemce automaticky doplní e-mailová adresa dodavatele. Nemá-li dodavatel v tabulce e-mail vyplněn, je uživateli nabídnuto doplnění adresy do formuláře s úpravou tohoto dodavatele. V případě, že uživatel e-mailovou adresu nedoplní, odesílání e-mailu uzavřeno a je zobrazena hláška: „Chyba při odesílání!“



Obrázek 17.: Odeslání objednávky e-mailem

4.4.7. Formuláře pro vytvoření a vypočtení nové mzdy

Karta *Nová mzda* je určena k výpočtu nové mzdy danému zaměstnanci za stávající období. Předtím než začne uživatel zadávat údaje k výpočtu mezd, musí pomocí roletky vybrat konkrétního zaměstnance, aby bylo jasné, komu je mzda počítána. Do polí *Jméno a Příjmení* jsou načteny aktuální hodnoty.

Tlačítkem *Vytvořit* je otevřeno okno, kde se automaticky doplní identifikace mzdy, složená z loginu a období. Dále období, skládající se z měsíce a roku a datum výpočtu.

K poslednímu kroku vytvoření mzdy je vymezeno tlačítko *Vypočítat*. Po kliknutí na něj dojde k otevření formuláře s poli, které vyplňuje pověřená osoba – pouze odpracovaná doba je povinná, a také ošetřena od nevyplnění. Ostatní pole, jako je např.: přesčas, odměna, osobní ohodnocení, ostatní srážky a příplatky a využitá dovolená, jsou vyplněny jen v případě, že s určitým zaměstnancem souvisí a jsou součástí výpočtu mzdy.

Naprogramovanou procedurou, která je přiložena v příloze 1, dojde k výpočtu mzdy k výplatě a jejich dílčích složek. Všechny tyto údaje jsou uloženy do tabulky *tblVypocteneMzdy* pod příslušnou identifikací. Následně jsou tyto hodnoty zobrazeny do polí, jež je možné vidět na obrázku 18.

Úvodní	Zboží	Nová objednávka	Přehled objednávek	Oddělení	Firma	Nová mzda	Přehled mezd
--------	-------	--------------------	-----------------------	----------	-------	-----------	-----------------

Vypočítat mzdu zaměstnanci

Přehled mezd

Vyberte login: Jméno: Příjmení:
 IDMzdy: Období: Datum výpočtu:

Pracovní doba	<input type="text"/>	Hrubá mzda	<input type="text"/>	Daň před slevou	<input type="text"/>
Přesčas	<input type="text"/>	Zdravotní hrazení	<input type="text"/>	Sleva celkem	<input type="text"/>
Odměna	<input type="text"/>	Socialní hrazení	<input type="text"/>	Daň po slevě	<input type="text"/>
Osobní ohodnocení	<input type="text"/>	Zdravotní placené	<input type="text"/>	Čistá mzda	<input type="text"/>
Nárok na dovolenou	<input type="text"/>	Socialní placené	<input type="text"/>	Ostatní srážky	<input type="text"/>
Využitá dovolená	<input type="text"/>			Ostatní příplatky	<input type="text"/>
				K výplatě	<input type="text"/>

Vypočítej

Obrázek 18.: Formulář pro vypočtení nové mzdy

Tlačítkem *Přehled mezd* je přepnut fokus na vedlejší kartu *Přehled mezd*. Pokud je zaměstnancem uzavřena nová mzda po vyplnění mzdových údajů, ale bez samotného výpočtu, jsou mzdové údaje z tabulky `tblMzdoveUdaje` odstraněny.

4.4.8. Formuláře pro přehled a úpravu mezd

Na kartě *Přehled mezd* je umožněno vybrat zaměstnance, jemuž je za určité období přiřazena mzda. Pomocí roletky s obdobím je možné vyhledat požadované období. Tlačítkem vedle data výpočtu, jak lze vidět na následujícím obrázku č. 19, se uživatel dostane do režimu úprav.

Vyberte zaměstnance

Období:

▶	IDMzdy	<input type="text" value="KUFO7895:4/2014"/>	Období	<input type="text" value="4/20"/>	Datum výpočtu	<input type="text" value="19. dubna 2014"/>	
	IDMzdy	<input type="text" value="KUFO8526:4/2014"/>	Období	<input type="text" value="4/20"/>	Datum výpočtu	<input type="text" value="19. dubna 2014"/>	
	IDMzdy	<input type="text" value="MART9888:4/2014"/>	Období	<input type="text" value="4/20"/>	Datum výpočtu	<input type="text" value="19. dubna 2014"/>	
	IDMzdy	<input type="text" value="RUCK6352:4/2014"/>	Období	<input type="text" value="4/20"/>	Datum výpočtu	<input type="text" value="18. dubna 2014"/>	

Obrázek 19.: Formulář s přehledem zaměstnanců

Pomocí sestavy byla vytvořena výplatní páska, kde stisknutím tlačítka *Tisk období* je zobrazen náhled tisku všech záznamů ve vybraném období, s následnou možností jejich vytisknutí. Podmínkou tohoto tlačítka je, že období musí být v roletce vybráno. Výplatní páska je přílohou č. 3.

Na obrázku 20 jsou demonstrovány výsledky výpočtu, ale také základní údaje o zaměstnanci – jméno, příjmení, rodné číslo, profese a hodinová sazba.

V případě chybného zadání vstupních údajů k výpočtu, je možné využít tlačítko *Upravit*, jímž je zobrazen formulář pro úpravu. Do polí jsou načteny výchozí hodnoty, které lze opravit a znovu potvrzením tlačítka provést výpočet.

Tlačítkem *Zpět* se uživatel dostane zpátky do výběru zaměstnanců, kde má možnost výběru jiného zaměstnance.

Tlačítkem *Tisk* ve formuláři přehledů mezd je odfiltrován pouze vybraný zaměstnanec a v náhledu tisku je zobrazena pouze jeho páska, kterou je možné vytisknout.

Přehled výpočtu					
IDMzdy	RUCK6352:4/2014	Období	4/20	Datum výpočtu	14. dubna 2014
Jméno	Pavla	Rodné číslo	702589/6352		
Příjmení	Rucká	Profese	Prodavač/ka		
		Hodinová sazba	65,00 Kč		
Pracovní doba	140	Hrubá mzda	9 100,00 Kč	Daň před slevou	1 830,00 Kč
Přesčas	0	Zdravotní hrazení	819,00 Kč	Sleva celkem	3 415,00 Kč
Odměna	0,00 Kč	Socialní hrazení	2 275,00 Kč	Daň po slevě	0,00 Kč
Osobní ohodnocení	0,00 Kč	Zdravotní placené	410,00 Kč	Čistá mzda	8 099,00 Kč
Nárok na dovolenou	40	Socialní placené	591,00 Kč	Ostatní srážky	0,00 Kč
Využitá dovolená	0			Ostatní příplatky	0,00 Kč
				K výplatě	8 099,00 Kč

Obrázek 20.: Formuláře s přehledem výpočtu

4.4.9. Formuláře pro evidenci a správu zaměstnanců

Karta *Zaměstnanci* je určena pro evidování jednotlivých zaměstnanců, úpravu stávajících údajů a přidávání nového zaměstnance.

Na obrázku 21 je vidět, že jsou zaznamenávány osobní, pracovní, přihlašovací údaje a údaje o slevách na daních. K selekci konkrétního zaměstnance může posloužit roletka, která po aktualizaci vyfiltruje zaměstnance, příp. lze použít šipky k posouvání na další nebo předchozí záznam.

Seznam zaměstnanců a jejich informace

Osobní údaje

Jméno: * Ulice:

KUFO7895 Kuřová Monika

KUFO8526 Kuřová Jana

MART9888 Martynková Karin

RUCK6352 Rucká Pavla

Příjmení: *

Rodné číslo: *

Telefon:

Mobil:

Město: *

PSČ: *

Pracovní údaje

Profese: ▼

Hod. sazba:

Právo: ☐

Údaje o slevách na dani

Poplatník: ☒

Počet dětí: ▼

PocetDetiZTP: ▼

Invalida 1. a 2. st.: ☐

Invalida 3. st.: ☐

ZTP: ☐

Student: ☐

Přihlašovací údaje

Přihlaš. jméno:

Heslo:

Obrázek 21.: Formulář s evidencí zaměstnanců

Formulář pro přidávání a úpravu zaměstnanců vzhledově odpovídá základnímu formuláři pro evidenci zaměstnanců. Přidávání záznamů je ošetřeno tak, aby každý zaměstnanec měl své jedinečné přihlašovací jméno, v opačném případě je vyhozena výjimka. Pole, která musí být vyplněna, jsou rovněž ošetřena proti nevyplnění a je na tyto pole graficky upozorněno „*“. Pokud zaměstnanec nevyplní nepovinné pole, je do něj automaticky vložena hodnota Null, příp. 0, je-li pole číselného formátu.

Při úpravě aktuálního zaměstnance jsou při otevření do formuláře pro úpravu nastaveny výchozí hodnoty, které zde zaměstnanec může upravovat. Opět je dbáno na stejný login i nevyplněné údaje.

K odstranění záznamu je vždy vyžadováno potvrzení, zda má být záznam odstraněn.

V příloze č. 4 je možné vidět přehled některých zmíněných formulářů jako např.: *Úprava sazebníku mezd*, *Úprava firemních údajů*, *Seznam oddělení*. Kompletní aplikace je dostupná na přiloženém externím médiu bakalářské práce.

Osobní údaje použity v této bakalářské práci byly záměrně pozměněny tak, aby nedošlo k jejich zneužití.

5. Závěr

Cílem této bakalářské práce bylo navrhnout databázovou aplikaci a prakticky ji zrealizovat. Práce byla zpracovávána pro maloobchodní jednotku Esíčko, kdy majitelé vystupují pod názvem ČeSi, s. r. o. Cíle zpracování byly stanoveny po zjištění všech požadavků. Aplikace měla být zaměřena na evidenci zboží, zaměstnanců a dodavatelů, dále na objednávací a mzdový systém.

Prvním úkolem bylo prozkoumat procesy v podniku a vytyčit si hlavní objekty. Tyto objekty převést do tabulek MS Accessu a určit jejich vztahy. Nakonec byla za pomoci dotazů, formulářů a sestav, vytvořena jednoduchá, přehledná a fungující aplikace.

Procesy, které se v podniku odehrávají, jsou mnohem sofistikovanější než samotná aplikace, ale snahou této práce bylo zahrnout v aplikaci ty nejdůležitější a vytvořit zaměstnancům příjemné uživatelské rozhraní. Např.: vytvoření nové objednávky, postupné přidávání objednaného zboží, vrácení se k rozpracované objednávce, možnost ji vytisknout nebo odeslat e-mailem. Rovněž práce s funkcemi správa zaměstnanců, počítání jeho mzdy za aktuální období, příp. jeho oprava, tisk výplatní pásky jednomu zaměstnanci nebo všem za celé období je velice intuitivní.

Všechny výše zmíněné systémy se podařilo spojit do jedné funkční aplikace, kterou by mohl být nahrazen původní nevyhovující systém.

Aplikace by mohla být implementována do praxe a připravena k odzkoušení. Ve zkušebním provozu se mohou objevit případné nedostatky, které budou muset být odstraněny. Případným rozšířením aplikace by mohlo být měsíční plánování směn každému zaměstnanci, tím i zjištění skutečně odpracované doby a počtu hodin, kterých zaměstnanec v daném měsíci musí dosáhnout.

Seznam použité literatury

a) Tištěné dokumenty

CONOLLY, T., C. BEGG a R. HOLOWCZAK. *Mistrovství - databáze: profesionální průvodce tvorbou efektivních databází*. Brno: Computer Press, 2009. ISBN 978-802-5123-287.

HERNANDEZ, Michal, J. *Návrh databází*. Praha: Grada Publishing, 2006. ISBN 80-247-0900-7.

OPPEL, Andy. *SQL bez předchozích znalostí: průvodce pro samouky*. Brno: Computer Press, 2008. ISBN 978-80-251-1707-1.

PÍSEK, Slavoj. *Access 2010: podrobný průvodce*. Praha: Grada Publishing, 2011. ISBN 978-80-247-3653-2.

RIORDAN, Rebecca, M. *Vytváříme relační databázové aplikace*. Brno: Computer Press, 2000. ISBN 80-7226-360-9.

SHEPHERD, Richard. *Access VBA: výukový průvodce*. Brno: Computer Press, 2012. ISBN 978-80-251-3686-7.

SINGH, S. K. *Database Systems Concepts, Design and Applications*. India: Dorling Kindersley, 2009. ISBN 978-81-7758-567-4.

VYBÍHAL, Václav. *Mzdové účetnictví 2013: praktický průvodce*. Praha: Grada Publishing, 2013. ISBN

b) Elektronické dokumenty

JaknaOffice.cz. Datové typy a vlastnosti polí tabulek. In: *JaknaOffice.cz* [online]. 20.11.2011 [cit. 2014-02-25]. Dostupné z: <http://www.jaknaoffice.cz/9-access/36-access-2010/41-datove-typy-a-vlastnosti-poli-tabulek/>

HAUZAR, David. Tvorba databází v MySQL – I. In: *AbcLinuxu* [online]. 20.03.2003 [cit. 2014-02-12]. Dostupné z: <http://www.abclinuxu.cz/clanky/navody/tvorba-databazi-v-mysql-i>

KUČEROVÁ, Dagmar. Změny ve slevách na dani pro rok 2014. In: *Podnikatel.cz* [online]. 27.1.2014 [cit. 2014-03-11]. Dostupné z: <http://www.podnikatel.cz/clanky/zmeny-ve-slevach-na-dani-pro-rok-2014/>

MICROSOFT. Filtr: Omezení počtu zobrazených záznamů. In: *Office.com* [online]. © 2014 [cit. 2014-02-25]. Dostupné z: <http://office.microsoft.com/cs-cz/access-help/filtr-omezeni-poctu-zobrazenych-zaznamu-HA010037839.aspx>

NOVÁK, Vítězslav. Přednášky k předmětu: Databázové aplikace [online]. 2012.

[cit. 2014-03-04]. Dostupné z: <http://lms.vsb.cz/course/view.php?id=7699>

PROCHÁZKA, Jaroslav. Objektově orientované databáze. In: *Databázový svět* [online].

03.03.2004 [cit. 2014-02-05]. Dostupné z:

<http://www.dbsvet.cz/view.php?cisloclanku=2004030301>

SKŘIVAN, Jaromír. Databáze nejsou jen MySQL. In: *Interval* [online]. 05.01.2002 [cit.

2014-02-03]. Dostupné z: <http://interval.cz/clanky/databaze-nejsou-jen-mysql/>

Seznam zkratek

1NF – První normální forma

2NF – Druhá normální forma

3NF – Třetí normální forma

4NF – Čtvrtá normální forma

5NF – Pátá normální forma

BCNF – Boyce/Coddova normální forma

ČR – Česká republika

DB - Databáze

DBA – Správce databáze

DML – Data Manipulation Language

DPH – Daň z přidané hodnoty

FO – Fyzická osoba

GIS – Geografický informační systém

IČO – Identifikační číslo organizace

IS – Informační systém

OS – Operační systém

OSVČ – Osoba samostatně výdělečně činná

PO – Právnícká osoba

PSZ – Pojistné na sociální zabezpečení

SQL – Structured Query Language

SW – Software

SŘBD – Systém řízení báze dat

UML – Use Case Diagram

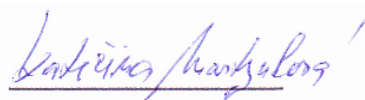
ZP – Zdravotní pojištění

Prohlášení o využití výsledků bakalářské práce

Prohlašuji, že

- jsem byla seznámena s tím, že na mou bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. – autorský zákon, zejména § 35 – užití díla v rámci občanských a náboženských obřadů, v rámci školních představení a užití díla školního a § 60 – školní dílo;
- beru na vědomí, že Vysoká škola báňská – Technická univerzita Ostrava (dále jen VŠB-TUO) má právo nevýdělečně, ke své vnitřní potřebě, bakalářskou práci užít (§ 35 odst. 3);
- souhlasím s tím, že bakalářská práce bude v elektronické podobě archivována v Ústřední knihovně VŠB-TUO a jeden výtisk bude uložen u vedoucího bakalářské práce. Souhlasím s tím, že bibliografické údaje o bakalářské práci budou zveřejněny v informačním systému VŠB-TUO;
- bylo sjednáno, že s VŠB-TUO, v případě zájmu z její strany, uzavřu licenční smlouvu s oprávněním užít dílo v rozsahu § 12 odst. 4 autorského zákona;
- bylo sjednáno, že užít své dílo, bakalářskou práci, nebo poskytnout licenci k jejímu využití mohu jen se souhlasem VŠB-TUO, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly VŠB-TUO na vytvoření díla vynaloženy (až do jejich skutečné výše).

V Ostravě dne 9. května 2014



Kateřina Martynková

Seznam příloh

- Příloha č. 1 – procedura k počítání mezd
- Příloha č. 2 – ukázka objednávky
- Příloha č. 3 – ukázka výplatního pásku
- Příloha č. 4 – přehled ostatních formulářů

Přílohy

Příloha č. 1

```
Private Sub btnOK_Click()  
    Dim log As String  
    Dim idm As String  
    Dim odprac As Long  
    Dim pres As Long  
    Dim odm As Long  
    Dim osohod As Long  
    Dim hru As Long  
    Dim zdrhr As Long  
    Dim sochr As Long  
    Dim zdrpl As Long  
    Dim socpl As Long  
    Dim danpr As Long  
    Dim popl As Long  
    Dim inv As Long  
    Dim inv3 As Long  
    Dim stud As Long  
    Dim zt As Long  
    Dim sleva As Long  
    Dim dan As Long  
    Dim cis As Long  
    Dim sr As Long  
    Dim prip As Long  
    Dim vyplata As Long  
    Dim narok As Long  
    Dim vyuz As Long  
    Dim dotaz As String  
  
    log = "'" & Login & "'"   
    idm = "'" & IDMzdy & "'"   
  
    If IsNull(OdpracDoba) Then  
        MsgBox "Vyplňte odpracovanou dobu!", vbInformation  
        OdpracDoba.SetFocus  
        Exit Sub  
    End If  
  
    odprac = OdpracDoba  
    If IsNull(Prescas) Then  
        pres = 0  
    Else  
        pres = Prescas  
    End If  
  
    If IsNull(Odmena) Then  
        odm = 0  
    Else  
        odm = Odmena  
    End If  
  
    If IsNull(OsOhodnoceni) Then  
        osohod = 0  
    Else  
        osohod = OsOhodnoceni  
    End If
```

```

hru = odprac * Form_frmMzdy.HodSazba + _
      (pres * Form_frmMzdy.HodSazba * 0.25) + odm + osohod
zdrhr = hru * Form_frmUpravitiSazebnik.SazbaZdravotniHr
sochr = hru * Form_frmUpravitiSazebnik.SazbaSocialniHr
zdrpl = hru * Form_frmUpravitiSazebnik.SazbaZdravotniPl
socpl = hru * Form_frmUpravitiSazebnik.SazbaSocialniPl
danpr = (Math.Round((hru + zdrhr + sochr) / 100) * 100) * _
      Form_frmUpravitiSazebnik.SazbaDan

If Form_frmMzdy.Poplatnik = 0 Then
    popl = 0
Else
    popl = Form_frmUpravitiSazebnik.SazbaPoplatnik
End If

If Form_frmMzdy.Invalida1_2 = 0 Then
    inv = 0
Else
    inv = Form_frmUpravitiSazebnik.SazbaInvalida1_2
End If

If Form_frmMzdy.Invalida3 = 0 Then
    inv3 = 0
Else
    inv3 = Form_frmUpravitiSazebnik.SazbaInvalida3
End If

If Form_frmMzdy.Student = 0 Then
    stud = 0
Else
    stud = Form_frmUpravitiSazebnik.SazbaStudent
End If

If Form_frmMzdy.ZTP = 0 Then
    zt = 0
Else
    zt = Form_frmUpravitiSazebnik.SazbaZTP
End If

sleva = popl + inv + inv3 + stud + zt

If (danpr - sleva < 0) Then
    dan = 0 - (Form_frmMzdy.PocetDeti * _
      Form_frmUpravitiSazebnik.SazbaDite + _
      Form_frmMzdy.PocetDetiZTP * _
      Form_frmUpravitiSazebnik.SazbaDiteZTP)
Else
    dan = danpr - sleva - Form_frmZamestnanci.PocetDeti * _
      Form_frmUpravitiSazebnik.SazbaDite + _
      Form_frmZamestnanci.PocetDetiZTP * _
      Form_frmUpravitiSazebnik.SazbaDiteZTP
End If

cis = hru - zdrpl - socpl - dan

If IsNull(OstSrazky) Then
    sr = 0
Else
    sr = OstSrazky
End If

```

```

If IsNull(OstPripl) Then
    prip = 0
Else
    prip = OstPripl
End If

vyplata = cis - sr + prip

If IsNull(VyuzDov) Then
    vyuz = 0
Else
    vyuz = VyuzDov
End If
    narok = 20 - vyuz

dotaz = "INSERT INTO tblVypocteneMzdy(Login, IDMzdy, OdpracDoba, "& _
"Prescas, Odmena, OsOhodnoceni, Hrubá, ZdravotniHr, SocialniHr, "& _
"ZdravotniPl, SocialniPl, DanPred, SlevaCelkem, DanPo, Cista, "& _
"OstSrazky, OstPripl, Kvyplate, NarokDov, VyuzDov) " & _
"VALUES(" & log & "," & idm & "," & odprac & "," & pres & "," & _
"" & odm & "," & osohod & "," & hru & "," & zdrhr & "," & _
"" & sochr & "," & zdrpl & "," & socpl & "," & danpr & "," & _
"" & sleva & "," & dan & "," & cis & "," & sr & "," & prip & "," & _
"" & vyplata & "," & narok & "," & vyuz & ")"

DoCmd.SetWarnings (False)
DoCmd.RunSQL (dotaz)
DoCmd.SetWarnings (True)
Form_frmPocitaniMezd.Requery
DoCmd.Close acForm, "frmPridatVypoctenouMzdu"
End Sub

```


Příloha č. 2

Objednávka č.: 56

Objednávka č.: 56

Datum vyhotovení: 7.4.2014
 Vyhotovil: KUFO7895
 Jméno: Monika
 Příjmení: Kuřová
 Telefon/Mobil:
 Forma platby: Hotově

Objednavatel: ČeSi, s. r. o. Mosty u Jablunkova, 1025
 Mosty u Jablunkova 739 98
 Telefon/Mobil: 558 367 611
Dodavatel: Hruška Táboristů 1
 Olomouc 779 00
 IČO: 11111111
 Telefon/Mobil: 585 119 135 789 652 365
 Fax: 585 119 135
 Email: mohruska@volny.cz

Kód zboží	Název	Množ	Jedn	Orientační cena	DPH
6	Jogurt	50	ks	8,00 Kč	15%
1	Rajče	200	ks	20,00 Kč	15%
Celková cena bez DPH: 4 400,00 Kč					

Razítko a podpis objednavatele

Stránka 1 z 1

Příloha č. 3

Mzda č.: RUCK6352:4/2014

CESI1025 ČeSi, s. r. o.
Mosty u Jablunkova, 1025 Mosty u Jablunkova 739 98 558 367 611

ID Mzdy RUCK6352:4/2014
Za období: 4/20
Datum výpočtu: 14. dubna 2014

Výplatní páska

RUCK6352	Pavla	Rucká	702589/6352	Prodavač/ka	65,00 Kč
----------	-------	-------	-------------	-------------	----------

OdpracDoba	140	Hrubá mzda	9 100,00 Kč	Daň před slevou	1 830,00 Kč
Přesčas	0	Zdravotní hrazení	819,00 Kč	Sleva celkem	3 415,00 Kč
Odměna	0,00 Kč	Socialní hrazení	2 275,00 Kč	Daň po slevě	0,00 Kč
Osobní ohodnocení	0,00 Kč	Zdravotní placené	410,00 Kč	Čistá mzda	8 099,00 Kč
Zbývajíc nárok	40	Socialní placené	591,00 Kč	Ostatní srážky	0,00 Kč
Využitá dovolená	0			Ostatní příplatky	0,00 Kč
				K výplatě	8 099,00 Kč

Seznam oddělení

Úprava firemních údajů

2	Ovoce	15%	<div><div>CES1025</div><div>Firma</div><div>ČeSi, s. r. o.</div></div>
3	Zelenina	15%	
4	Maso/uzeniny	15%	
5	Mléčné výrobky	15%	
6	Káva/čaj	15%	
9	Těstoviny	15%	
10	Sladkosti/slané	15%	
11	Nealkoholické nápoje	15%	
12	Pěčivo	15%	
13	Drogérie	21%	
14	Elektro	21%	
15	Železářství	21%	
16	Barvy/laky	21%	
17	Zdravotnické potřeby	15%	

Přidat

Upravit

Odstranit

KodFirmy

Bankovní údaje:

Kontaktní údaj:

Banka

Bankovní údaje:

Mosty u Jablunkova, 1025

Banka

Mosty u Jablunkova

ČeSi, s. r. o.

739 98

Předčísí

1234567899

000000

558 367 611

Číslo účtu

0100

000000

Kód banky

Úprava mzdových sazeb

Poplatník

Dítě

Dítě ZTP

Invalidní 1. a 2. stupeň

Invalidní 3. stupeň

ZTP

Student

2 070,00 Kč

1 117,00 Kč

1 934,00 Kč

210,00 Kč

420,00 Kč

1 345,00 Kč

335,00 Kč

Zdravotní hrazené

Sociální hrazené

Super hrubá mzda

Daň z příjmů

Zdravotní placené

Sociální placené

9,00%

25,00%

34%

15%

4,50%

6,50%